Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

RGB--Infrared Photovoltaic Defect Detection via Edge-Aware Feature Extraction and Multi-Scale Fusion

Chen Zhao
Hebei University of Architecture
Dongxu Liu
Hebei University of Architecture
Shijie Feng
Hebei University of Architecture
Jiahui Li
Hebei University of Architecture
Hanshuo Yuan
Hebei University of Architecture
Bosong Li
Hebei University of Architecture
Yunchang Zheng
Hebei University of Architecture
Qing Chang
cq1066@hebiace.edu.cn

Hebei University of Architecture

Research Article

Keywords:

Posted Date: November 27th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11751032/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

RGB–Infrared Photovoltaic Defect Detection via Edge-Aware Feature Extraction and Multi-Scale Fusion

Abstract

Reliable photovoltaic (PV) defect detection is essential for maintaining solar power system performance. However, variations in defect appearance, loss of fine-grained spatial details, and limited interaction between visible and thermal features make accurate detection challenging. To address these issues, this study proposes Photovoltaic Multi-modal Feature-enhanced YOLO (PVMF-YOLO), a YOLO11n-based RGB–infrared detector. It incorporates three modifications: a Laplacian-of-Gaussian stem (LoGStem) in the RGB backbone to enhance shallow structural features; P3 cross-modal fusion and an additional P2 detection branch to improve high-resolution feature representation; and an SOEP–RFPN neck for multi-scale feature aggregation. Experiments on the six-category DM-PV dataset show that PVMF-YOLO achieves a Precision of 0.868, Recall of 0.862, F1-score of 0.865, mAP_{50} of 0.906, and $mAP_{50:95}$ of 0.645 on the independent test set. The proposed model improves Precision, Recall, F1-score, mAP50, and mAP50:95 by 6.5, 5.3, 5.9, 5.5, and 5.4 percentage points, respectively, compared with the dual-modal YOLO11n-MM baseline. All seven YOLO architectures achieve higher values for both mAP metrics when using dual-modal inputs than when using RGB or infrared inputs alone. The ablation results show that combining P3 fusion with P2 detection brings the largest improvements in mAP. The effect of SOEP–RFPN varies depending on the other modifications used with it. These improvements also require more computation, showing a trade-off between detection performance and model complexity.

Keywords: Multi-scale feature enhancement, photovoltaic defect detection, PVMF-YOLO, RGB–infrared fusion, small object detection

1 Introduction

As the number of photovoltaic (PV) systems increases, the need for effective inspection and maintenance also grows. Problems such as local overheating, surface damage, and cell abnormalities can reduce power generation and affect system reliability. Manual inspection of large PV installations requires considerable time and effort, making it

difficult to apply on a large scale. Deep learning-based methods can help automate PV inspection (Akram et al. 2022; Høiaas et al. 2022; Li et al. 2021; Rahman et al. 2021).

Visible-light, infrared (IR), and electroluminescence (EL) imaging are commonly used for PV inspection (Høiaas et al. 2022; Sabry et al. 2026; Yahya et al. 2022). Visible-light images show surface texture and structural details, while IR images provide thermal information that can help identify potential faults. These advantages have led to studies on visible-light defect recognition (Cao et al. 2023; Ramachandran et al. 2025; Zhu et al. 2026) and RGB–IR multimodal detection (Hong et al. 2022; Kuo et al. 2023; Lai et al. 2025). However, each modality has its own limitations. Visible-light images can be affected by changes in illumination and cluttered backgrounds, while IR images often contain less structural detail, making precise defect localization more difficult. Combining RGB and IR information can help address these limitations by using both visual and thermal features.

Multimodal imaging provides complementary information, but PV defect detection remains challenging. Defects differ in size and appearance, and their thermal responses also vary. Small defects and low-contrast targets are often difficult to distinguish from the surrounding areas. Repeated downsampling further reduces the spatial detail available for localization. Although multi-scale feature aggregation can combine shallow and deep representations, it may not fully recover information lost during feature extraction. Introducing high-resolution detection branches and feature enhancement modules also increases computational cost (Gong et al. 2025; Mohanty et al. 2025; Su et al. 2022; Zhang et al. 2024b). Figure 1 shows common PV inspection challenges and explains why RGB and IR features are combined.



Fig. 1 Motivation and challenges of RGB–infrared PV defect detection

Research on RGB–infrared detection has explored cross-modal attention, feature alignment, complementary feature learning, and redundancy suppression (Fang and Wang 2022; Shen et al. 2024; Wang et al. 2022b; Yuan and Wei 2024; Yuan et al. 2024). Related fusion strategies have also been investigated in other RGB–thermal recognition tasks (Gao et al. 2022; Song et al. 2023; Wang et al. 2023). In PV inspection, Kang et al. used synergistic graph-contrastive regularization for multimodal anomaly detection (Kang et al. 2026), while FAF-YOLO addressed cross-modal misalignment and variations in defect scale in UAV imagery (Wang et al. 2026). These studies provide a basis for exploiting complementary modalities. For small PV defects, however, effective fusion also depends on retaining local structural information and carrying it through the detection hierarchy. This motivates a coordinated design of shallow feature extraction, high-resolution cross-modal fusion, and multi-scale aggregation.

This study presents PVMF-YOLO (Photovoltaic Multi-modal Feature-enhanced YOLO), a dual-modal RGB–infrared detector based on YOLO11n. The proposed architecture incorporates a LoGStem module in the RGB backbone, explicit cross-modal fusion at P3 together with an additional P2 detection branch, and an SOEP–RFPN neck for multi-scale feature aggregation. We conduct comparative experiments and ablation studies to examine the effects of these modifications on detection performance and computational cost, both separately and in combination.

This study makes the following contributions:

1. The proposed model adds cross-modal fusion at P3 and an additional P2 detection branch. The detection scales are extended from P3–P5 to P2–P5. This allows RGB and infrared features to interact at a higher resolution and provides more spatial details for detecting small defects.
2. The RGB backbone uses a Laplacian-of-Gaussian stem (LoGStem). It combines edge-sensitive filtering with Gaussian-based feature enhancement to retain local structural details in the early stages of feature extraction.
3. The SOEP–RFPN neck is used to combine high-resolution spatial features with deeper semantic features across different scales. Ablation experiments assess its effects on detection performance and computational complexity, including its interaction with the other architectural modifications.

2 Related Work

This section reviews three research areas relevant to the proposed method: deep learning-based PV inspection, RGB–infrared feature fusion, and high-resolution multi-scale feature learning. Figure 2 summarizes these areas and their connections to the motivation of this study.

2.1 Deep Learning for Photovoltaic Inspection

Deep learning methods have been applied to PV inspection using infrared, visible-light, and electroluminescence (EL) images, with each modality supporting different inspection tasks. Infrared-based studies primarily address thermal fault recognition and anomaly detection. Akram et al. explored transfer learning for fault recognition

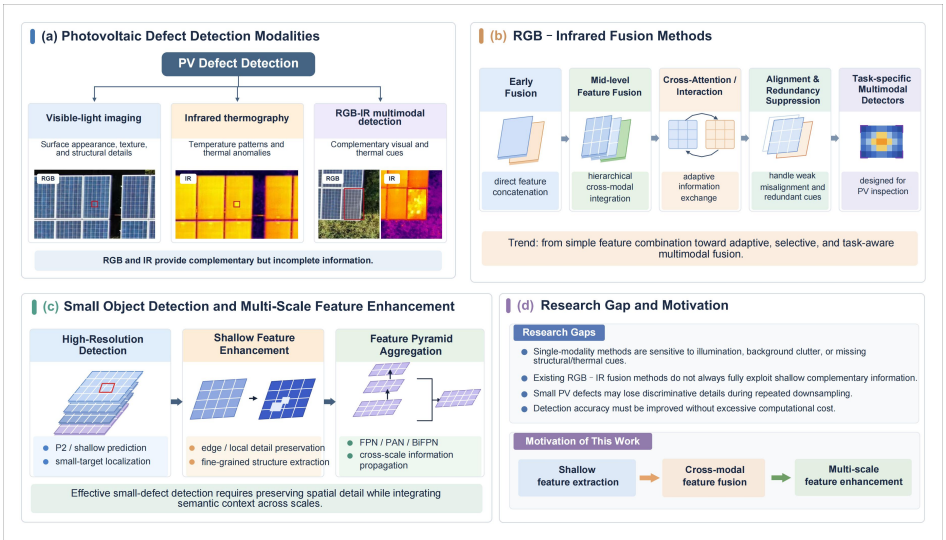


Fig. 2 Overview of related research: (a) imaging modalities for PV inspection; (b) RGB–infrared fusion strategies; (c) high-resolution and multi-scale feature learning; and (d) research gaps and motivation

in infrared images, while Mellit and Kellil et al. investigated thermographic diagnosis under different operating and climatic conditions (Akram et al. 2020; Kellil et al. 2023; Mellit 2022). Further work has used supervised contrastive learning to detect thermal anomalies and introduced high-resolution UAV datasets, including PVF-10, to support fine-grained fault classification (Bommes et al. 2022; Wang et al. 2024b).

Visible-light methods identify defects from surface appearance, texture, and structural details. Research in this area has examined image preprocessing and feature extraction for PV defect detection and classification (Cao et al. 2023; Ramachandran et al. 2025; Zhu et al. 2026). In solar-cell manufacturing inspection, BAF-Detector and complementary attention networks have explored efficient convolutional architectures and attention-based feature enhancement (Su et al. 2021, 2022). However, differences in image acquisition conditions and defect types limit their use in outdoor PV inspection.

EL images can show internal cell abnormalities and manufacturing defects. Previous studies have used them for defect detection, anomaly recognition, and fault diagnosis (Korovin et al. 2023; Ozturk et al. 2024; Wang et al. 2021). To improve detection efficiency and feature representation, researchers have explored lightweight models based on neural architecture search and knowledge distillation, as well as multi-scale feature extraction methods (Fan et al. 2026; Zhang et al. 2024a). The acquisition requirements of EL imaging nevertheless differ from those of UAV-based RGB and infrared inspection, limiting the direct transfer of these methods to aerial inspection settings.

Computational efficiency is another consideration in practical PV inspection. Lightweight detectors, FPGA acceleration, and edge–cloud collaborative processing have been investigated to meet different deployment requirements (Gong et al. 2025;

Tang et al. 2022; Wang et al. 2022a; Zhang et al. 2024b). Within this broader research landscape, RGB–infrared imaging offers a means of combining surface details with thermal cues for outdoor inspection. The following subsection reviews methods for fusing these complementary sources of information.

2.2 RGB–Infrared Fusion for Multimodal Detection

RGB–infrared detection combines visual and thermal cues to learn complementary feature representations (Song et al. 2023; Wang et al. 2023). Research in this area has examined feature-level fusion and cross-modal interaction, alongside methods for aligning heterogeneous features and reducing redundant information.

For PV inspection, Hong et al. combined PV array segmentation in visible-light images with infrared defect recognition, while Kuo et al. investigated UAV-based RGB–infrared defect detection and localization in large PV plants (Hong et al. 2022; Kuo et al. 2023). Kang et al. used synergistic graph-contrastive regularization for multimodal PV anomaly detection (Kang et al. 2026). These studies use visual and thermal information in different ways. Some methods combine features from separate branches for each modality, while others learn joint feature representations to detect anomalies.

Attention mechanisms have also been widely used to fuse features from different modalities beyond PV inspection. CMAFF uses differential-modality and common-modality attention to distinguish modality-specific cues from shared information (Fang and Wang 2022). ICAFusion integrates complementary features through iterative cross-attention (Shen et al. 2024). C^2Former uses cross-attention between modalities and adaptive feature sampling to improve feature interaction while reducing computational cost (Yuan and Wei 2024).

Researchers have paid close attention to feature misalignment and redundancy because both can make it difficult to combine RGB and infrared features. Yuan et al. proposed a cascade alignment-guided Transformer that combines geometric alignment with region-level feature fusion (Yuan et al. 2024). Wang et al. used mutual-information minimization and feature selection to suppress cross-modal redundancy (Wang et al. 2022b). Other studies have investigated thermal-guided integration, unified information fusion, relation-map difference-guided fusion, and modality-knowledge learning (Gao et al. 2022; Ma et al. 2026; Shen et al. 2026; Wang et al. 2023). These methods aim to preserve complementary information while reducing feature differences and redundancy between modalities.

FAF-YOLO was developed for UAV-based PV inspection. To address weak RGB–infrared misalignment, its Frequency-aware Selective Fusion (FSF) module separates low-frequency structural information from high-frequency details and selectively fuses features from the two modalities. The model also uses defect-oriented feature enhancement and a scale-differentiated detection head to handle differences in defect appearance and size. Since FAF-YOLO was evaluated on the DM-PV dataset, it provides a useful reference for the task and evaluation setting of this study (Wang et al. 2026).

These studies show the importance of selective interaction between different modalities. However, preserving fine spatial details throughout the detection network remains

difficult. For small PV defects, high-resolution features need to retain local details and include the semantic information required for recognition. Additional fusion and feature enhancement operations can also increase computational cost. These challenges motivate this study to investigate high-resolution cross-modal fusion and multi-scale feature aggregation.

2.3 High-Resolution and Multi-Scale Feature Learning

Multi-scale feature learning helps object detectors detect targets of different sizes and is also used in PV defect detection (Fan et al. 2026; Huang et al. 2026; Qu et al. 2024). Convolutional detectors extract features at different levels through repeated downsampling. Shallow layers retain local details, while deeper layers capture more semantic information. Effective multi-scale detection needs to preserve spatial details and combine features from different network stages.

High-resolution features are useful for detecting small defects because their spatial details may be lost during downsampling. A P2 detection branch with a stride of four provides more spatial details than the conventional P3–P5 detection branches. Previous studies have also used multi-scale features to detect fine-grained PV defects (Fan et al. 2026; Zhang et al. 2024b). However, high-resolution features require more computation and may retain background information that affects defect detection. Combining these features with semantic information can help improve detection performance.

Enhancing shallow features offers a complementary way to preserve local structures before they are weakened by downsampling. Multi-branch convolutions, attention mechanisms, and local detail aggregation have been used to strengthen feature representations in PV inspection (Qu et al. 2024; Su et al. 2021, 2022). Gradient- and Laplacian-based operators highlight changes in image intensity and object boundaries, while Gaussian smoothing helps reduce noise. However, edge-sensitive operations may also highlight irrelevant textures. Their outputs therefore need to be combined with other features to distinguish useful structural information from background details.

Feature pyramid networks combine features from different spatial resolutions. FPN uses a top-down path to transfer high-level semantic information to higher-resolution features. PANet adds a bottom-up path to improve information flow between different levels. BiFPN combines features in both directions through repeated connections and uses learnable weights to adjust the contribution of each feature level (Lin et al. 2017; Liu et al. 2018; Tan et al. 2020). These architectures allow features at different scales to interact. However, adding more fusion paths does not always improve feature representation, especially when the transferred features contain redundant information.

Researchers have also used attention mechanisms, receptive-field adaptation, and contextual aggregation to improve multi-scale feature representations (Fan et al. 2026; Huang et al. 2026; Su et al. 2021). These methods provide different ways to combine local details with broader contextual information, but they may also increase computational cost. For PV defect detection, an effective model needs to combine high-resolution prediction, shallow feature enhancement, and cross-scale feature fusion while reducing feature redundancy and computational cost.

3 Materials and Methods

3.1 Dataset and Task Formulation

3.1.1 Dataset Description and Annotation

We evaluated PVMF-YOLO on the Dual-Modal Photovoltaic (DM-PV) dataset (Wang et al. 2026). The dataset was collected during real-world PV inspections using a DJI Matrice 3TD UAV. The original RGB and infrared images have resolutions of 5184 × 3888 and 640 × 512 pixels, respectively. The dataset authors applied center cropping with fixed parameters to the RGB images and resized both modalities to 1280 × 1024 pixels. As precise image registration was not performed, the released image pairs retain residual spatial misalignment. These images were used in this study without additional offline geometric processing.

The dataset comprises 2,375 RGB–infrared image pairs covering six PV defect categories. Representative samples are shown in Fig. 3.

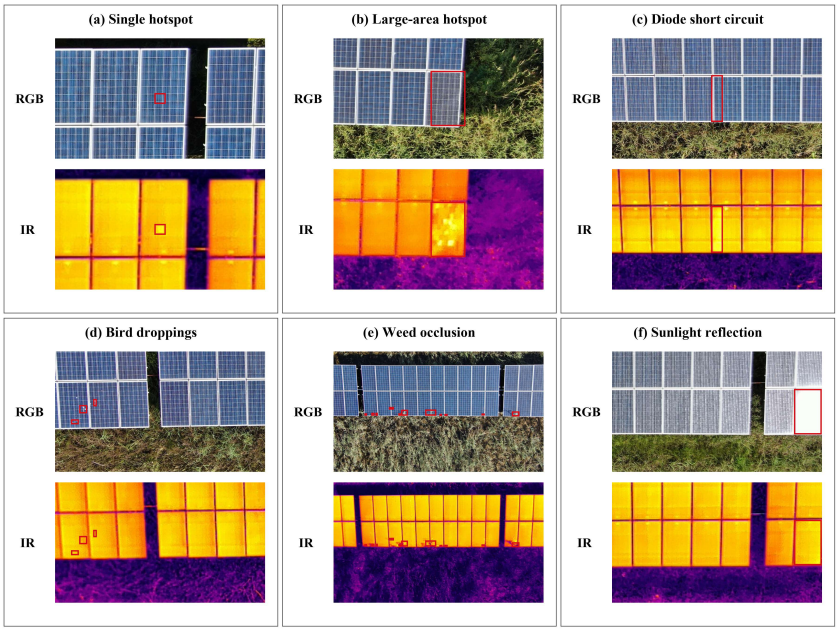


Fig. 3 Representative RGB–infrared image pairs for the six inspection categories in the DM-PV dataset

The infrared images are provided as three-channel pseudo-color JPEGs with an Ironbow-like color mapping. These images use color to show thermal differences, but they do not provide absolute temperature values or raw radiometric data.

Each RGB image has a corresponding infrared image with the same six-digit file identifier. File-level checks confirmed that all 2,375 image pairs match correctly. Each pair has one annotation file, and all bounding boxes use the same coordinate system of

1280 × 1024 pixels. Based on the original annotation protocol, hotspot-related defects and diode short circuits are mainly identified using infrared thermal information. Bird droppings, weed occlusion, and sunlight reflection are labeled based on visible features in RGB images (Wang et al. 2026).

3.1.2 Dataset Statistics, Partitioning, and Task Formulation

Table 1 lists the number of image pairs and bounding boxes for each category in the training, validation, and test sets. The "Images" column shows the number of paired samples containing at least one instance of each category. The Boxes" column gives the total number of annotated instances in each category.

Table 1 Class-wise distribution of image pairs and bounding-box instances in the DM-PV dataset

Defect Category	**Training**		**Validation**		**Test**	
	Images	Boxes	Images	Boxes	Images	Boxes
Single Hotspot	843	2,098	100	132	101	135
Large-area Hotspot	84	1,379	4	96	3	39
Diode Short Circuit	62	72	8	8	8	8
Bird Droppings	1,081	3,616	74	239	74	214
Weed Occlusion	591	1,933	65	369	65	378
Sunlight Reflection	88	226	11	29	11	25
Total	**1,900**	**9,324**	**238**	**873**	**237**	**799**

Note: The "Images" column shows the number of paired samples containing at least one instance of the corresponding category, while the "Boxes" column shows the total number of annotated instances. Image counts may overlap across categories because one sample can contain multiple defect types. The Total row gives the number of unique image pairs and the total number of bounding boxes in each subset.

The class distribution is highly imbalanced. Bird droppings account for 37.0% of all annotated instances, while diode short circuits account for only 0.8%. Thus, bird-dropping instances are about 46 times more common than diode short-circuit instances.

The dataset was randomly divided into training, validation, and test sets at the paired-sample level, with a ratio of approximately 8:1:1. Each RGB–infrared image pair and its annotation were assigned to the same subset. We also checked the file identifiers and confirmed that no image pair appeared in more than one subset.

Scene-level splitting was not possible because the released dataset does not provide usable GPS coordinates, timestamps, or flight-route identifiers. We also checked the file indices. Among the 2,255 pairs with adjacent file indices, 710 (31.5%) were assigned

to different subsets. To check for possible similarities across subsets, we calculated normalized cross-correlation using image thumbnails. The results showed that 397 cross-subset sample pairs had correlation values above 0.90, and 12 of them had values above 0.95. No pair had a value above 0.98. However, the absence of pairs above this threshold does not prove that the samples in different subsets come from independent scenes. Possible scene overlap between subsets therefore remains a limitation when interpreting the experimental results.

PV defect inspection was treated as a supervised object detection task. For each RGB–infrared image pair, the network outputs bounding boxes, class labels, and confidence scores, as shown in Eq. (1):

$$\hat{Y}^{(i)} = f_\theta \left(I_{\mathrm{RGB}}^{(i)}, I_{\mathrm{IR}}^{(i)} \right), \tag{1}$$

where $I_{\mathrm{RGB}}^{(i)}$ and $I_{\mathrm{IR}}^{(i)}$ denote the RGB and infrared images of the i-th sample, respectively, f_θ is the detection network with parameters θ, and $\hat{Y}^{(i)}$ denotes its predicted detections.

3.2 Baseline Dual-Modal YOLO11n Architecture

A dual-modal network built on YOLO11n, denoted as YOLO11n-MM, was used as the baseline to evaluate the proposed architectural modifications (Jocher and Qiu 2024). As illustrated in Fig. 4, it adopts a mid-level fusion design with parallel RGB and infrared backbones, cross-modal fusion at P4 and P5, an FPN–PAN neck, and a three-scale detection head.

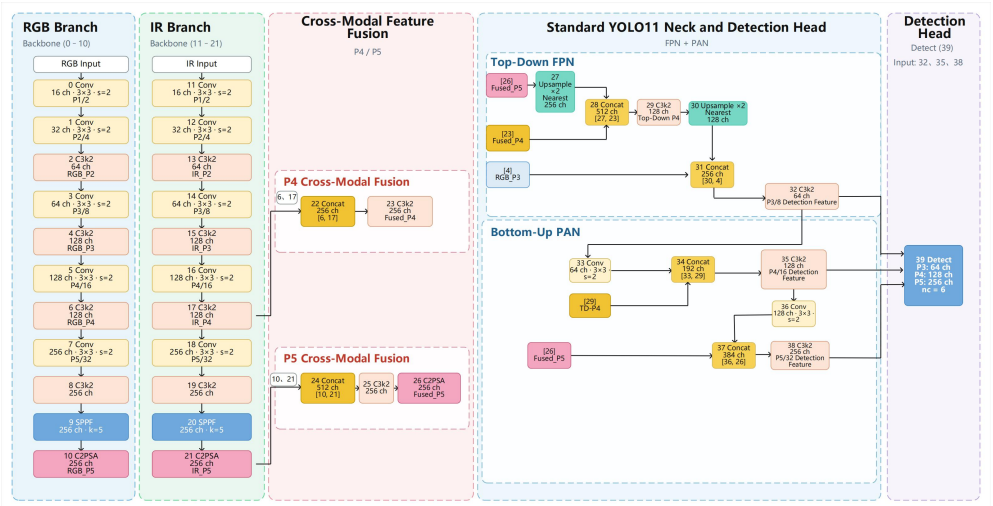


Fig. 4 Architecture of the baseline RGB–infrared dual-modal YOLO11n network (YOLO11n-MM)

The RGB and infrared branches extract features independently using the standard YOLO11n backbone structure, comprising convolutional layers, C3k2 blocks, Spatial

Pyramid Pooling-Fast (SPPF), and C2PSA modules. Each branch extracts features at P3, P4, and P5, with strides of 8, 16, and 32, respectively. At P4 and P5, the RGB and infrared features are concatenated along the channel dimension and then processed by C3k2 blocks. The fused P5 feature is further refined by a C2PSA module.

An FPN–PAN neck combines features at different scales (Lin et al. 2017; Liu et al. 2018). In the top-down path, the fused P5 feature is first upsampled and concatenated with the fused P4 feature. A C3k2 block then processes the combined features. The resulting P4 feature is upsampled and combined with the RGB P3 feature to form the P3 detection feature. In this way, cross-modal information is passed to P3 through the top-down path, but the RGB and infrared backbone features are not directly fused at this level. In the bottom-up path, features are passed from P3 to P5 through stride-2 convolutions, concatenation, and C3k2 blocks.

The detection head uses three scales: P3/8, P4/16, and P5/32. Although the baseline combines RGB and infrared features at the intermediate and deep levels, it retains conventional convolutional stems and does not include direct cross-modal fusion at P3 or an additional P2 detection branch. This baseline provides a reference for evaluating the shallow feature enhancement, high-resolution cross-modal fusion, and improved multi-scale feature aggregation introduced in PVMF-YOLO.

3.3 PVMF-YOLO for Photovoltaic Defect Detection

PVMF-YOLO builds on the dual-modal YOLO11n baseline with three architectural modifications: LoGStem-based shallow feature extraction, direct P3 cross-modal fusion coupled with an additional P2 detection branch, and SOEP–RFPN-based multi-scale feature aggregation. These modifications are designed to preserve local structural details and improve feature interaction across modalities and spatial scales.

3.3.1 Overall Architecture of PVMF-YOLO

Figure 5 presents the overall architecture of PVMF-YOLO. The network comprises parallel RGB and infrared backbones, cross-modal fusion at P3–P5, an SOEP–RFPN neck, and a four-scale detection head. The two backbones first extract modality-specific features, which are then fused and aggregated across scales.

The two backbones use different stem structures. In the RGB branch, LoGStem replaces the first two conventional convolutional layers. It combines edge-sensitive filtering, Gaussian enhancement, and downsampling to generate P2 features at a stride of four. The infrared branch retains the standard YOLO11n stem. Both branches subsequently extract P3, P4, and P5 features at strides of 8, 16, and 32, respectively, through convolutional layers and C3k2 blocks, with SPPF and C2PSA modules used at deeper stages.

The baseline model fuses RGB and infrared features only at P4 and P5. PVMF-YOLO adds direct feature fusion at P3. At each scale, the RGB and infrared features are concatenated along the channel dimension and passed through C3k2 blocks. The fused P5 features are further refined by a C2PSA module. Adding P3 fusion allows visual and thermal features to interact directly at a higher resolution. It also works with the cross-modal information passed from deeper layers.

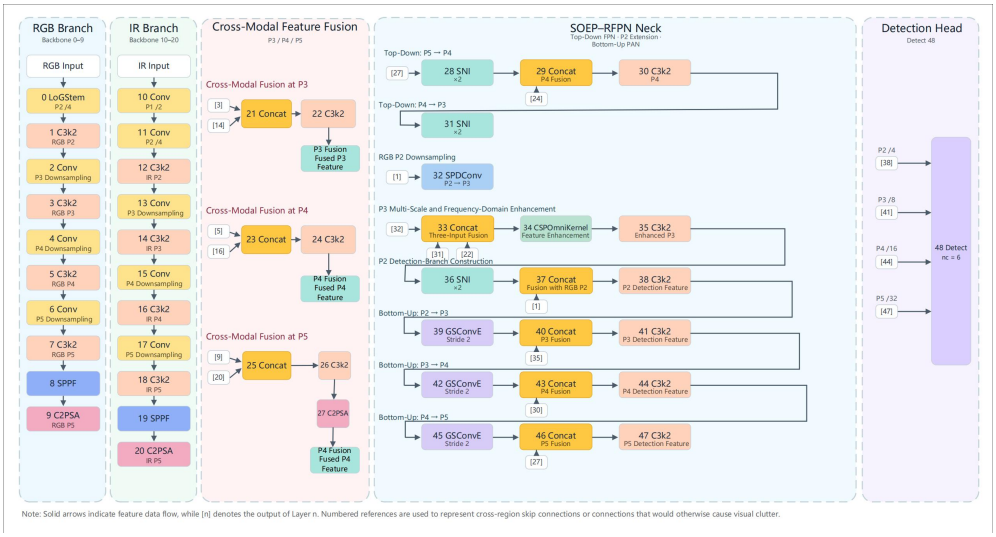


Fig. 5 Overall architecture of the proposed PVMF-YOLO network for RGB–infrared photovoltaic defect detection

The SOEP–RFPN neck replaces the conventional FPN–PAN structure and integrates SNI, SPDConv, CSPOmniKernel, and GSConvE modules. In the top-down pathway, SNI transfers features from P5 to P4 and then to P3. At P3, SPDConv-processed RGB P2 features are combined with upsampled P4 features and the fused P3 representation. The combined features are processed by CSPOmniKernel and C3k2 blocks to form an enhanced P3 representation. This representation is then upsampled and merged with the RGB P2 features to construct the additional P2 detection feature map. Along the bottom-up pathway, GSConvE-based downsampling and feature aggregation refine the P3–P5 detection representations.

The detection head takes four feature maps as input, corresponding to P2/4, P3/8, P4/16, and P5/32. Section 3.3.3 explains the P3 fusion and P2 detection mechanisms, and Section 3.3.4 describes the SOEP–RFPN neck. Comparative experiments and ablation studies are used to evaluate these modifications.

3.3.2 LoGStem-Based Edge-Aware Feature Extraction

A Laplacian-of-Gaussian-based stem (LoGStem) is introduced into the RGB backbone to preserve local visual details during early downsampling. The module integrates edge-sensitive filtering, convolutional downsampling, Gaussian residual enhancement, and dual-branch feature aggregation. As illustrated in Fig. 6, processing proceeds through four stages: LoGFilter, Conv_D, Gaussian residual enhancement, and DRFD_LoG. LoGStem replaces the conventional RGB stem, while the infrared branch retains the standard YOLO11n stem.

For an RGB input $\mathbf{I} \in \mathbb{R}^{B\times 3\times H\times W}$, a 7×7 convolution with stride 1 and padding 3 first produces a feature map at the original spatial resolution, as shown in Eq. (2):

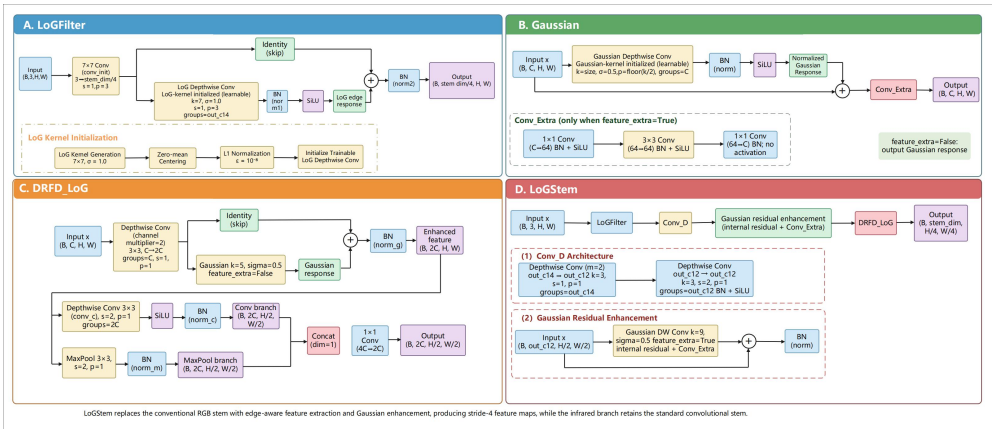


Fig. 6 Structure of LoGStem: (A) LoGFilter, (B) Gaussian module, (C) DRFD_LoG, and (D) the complete LoGStem architecture

$$\mathbf{F}_0 = \mathrm{Conv}_{7\times7,s=1,p=3}(\mathbf{I}), \qquad \mathbf{F}_0 \in \mathbb{R}^{B\times C_1\times H\times W}, \tag{2}$$

where B is the batch size and $C_1 = \text{stem_dim}/4$ is the intermediate channel dimension.

LoGFilter applies a learnable depthwise convolution initialized with a Laplacian-of-Gaussian (LoG) kernel. The LoG operator combines Gaussian smoothing with second-order differentiation to capture local intensity variations and structural boundaries (Marr and Hildreth 1980). Its scale-normalized form is given in Eq. (3):

$$\begin{aligned} K_\sigma^{(0)}(u,v) &= \sigma^2\nabla^2 G_\sigma(u,v) \\ &= \frac{u^2+v^2-2\sigma^2}{2\pi\sigma^4}\exp\left(-\frac{u^2+v^2}{2\sigma^2}\right), \end{aligned} \tag{3}$$

where G_σ is the two-dimensional Gaussian function, $(u,v) \in \Omega_7$, $\Omega_7 = \{-3,-2,\ldots,3\}^2$, and $\sigma = 1.0$. At this value of σ, the scale-normalized kernel is numerically identical to the conventional LoG kernel. The sampled kernel is then mean-centered and L1-normalized according to Eq. (4):

$$\begin{aligned} K_\sigma^{(c)}(u,v) &= K_\sigma^{(0)}(u,v) - \frac{1}{49}\sum_{(i,j)\in\Omega_7} K_\sigma^{(0)}(i,j), \\ K_\sigma^{(n)}(u,v) &= \frac{K_\sigma^{(c)}(u,v)}{\sum_{(i,j)\in\Omega_7}\left|K_\sigma^{(c)}(i,j)\right| + \varepsilon}. \end{aligned} \tag{4}$$

Here, $\varepsilon = 10^{-8}$ ensures numerical stability. The normalized kernel is replicated across the C_1 channels to initialize a 7×7 depthwise convolution. Each channel has its own trainable kernel, allowing the filters to adjust during training based on their initial edge-sensitive weights.

The depthwise convolution output is passed through batch normalization and SiLU activation before being added to $\mathbf{F}_0$. The output of the second batch normalization layer is given in Eq. (5):

$$\mathbf{F}_L = \mathrm{BN}_2\left(\mathbf{F}_0 + \mathrm{SiLU}\left(\mathrm{BN}_1\left(\mathrm{DWConv}_{\theta_L}(\mathbf{F}_0)\right)\right)\right), \tag{5}$$

Here, θ_L denotes the learnable weights initialized with $K_\sigma^{(n)}$. The residual connection adds edge-sensitive features to the input while keeping the feature dimensions unchanged.

A module with Gaussian-initialized weights provides another way to process local features. The discrete Gaussian kernel and its normalization are given in Eq. (6):

$$\begin{aligned} G_\sigma^{(0)}(u,v) &= \frac{1}{2\pi\sigma^2}\exp\left(-\frac{u^2+v^2}{2\sigma^2}\right), \\ \widehat{G}_\sigma(u,v) &= \frac{G_\sigma^{(0)}(u,v)}{\displaystyle\sum_{(i,j)\in\Omega_k} G_\sigma^{(0)}(i,j)}, \end{aligned} \tag{6}$$

where Ω_k denotes the $k\times k$ kernel support. The normalized kernel initializes a learnable depthwise convolution. For an input feature map $\mathbf{X}$, the module computes the response in Eq. (7):

$$\mathbf{R}_G(\mathbf{X}) = \mathrm{SiLU}\left(\mathrm{BN}_G\left(\mathrm{DWConv}_{\theta_G}(\mathbf{X})\right)\right), \tag{7}$$

where θ_G denotes the Gaussian-initialized convolutional weights. The `feature_extra` setting determines whether this response is returned directly or combined with the input and processed by an additional convolutional block, as specified in Eq. (8):

$$\mathrm{Gaussian}_{k,\sigma}(\mathbf{X}) = \begin{cases} \mathrm{Conv_Extra}(\mathbf{X} + \mathbf{R}_G(\mathbf{X})), \\ \qquad \text{feature_extra} = \text{True}, \\ \mathbf{R}_G(\mathbf{X}), \\ \qquad \text{feature_extra} = \text{False}. \end{cases} \tag{8}$$

Conv_Extra consists of three convolutional layers with channel dimensions $C \to 64 \to 64 \to C$. The first 1×1 convolution and the subsequent 3×3 convolution each use batch normalization and SiLU activation; the final 1×1 convolution uses batch normalization only. LoGStem uses a 9×9 kernel with `feature_extra=True` in the main enhancement pathway and a 5×5 kernel with `feature_extra=False` in DRFD_LoG. Both configurations use $\sigma = 0.5$.

Following LoGFilter, Conv_D expands the channel dimension and performs the first spatial downsampling. Let $C_2 = \text{stem_dim}/2 = 2C_1$. A 3×3 depthwise convolution with a channel multiplier of two expands the features from C_1 to C_2 channels while preserving spatial resolution. A second depthwise convolution with stride 2 then downsamples the features, as formulated in Eq. (9):

$$\begin{aligned} \mathbf{F}_{D1} &= \mathrm{DWConv}_{3\times 3, s=1, g=C_1}^{C_1\to C_2}(\mathbf{F}_L), \\ \mathbf{F}_D &= \mathrm{SiLU}\left(\mathrm{BN}_D\left(\mathrm{DWConv}_{3\times 3, s=2, g=C_2}(\mathbf{F}_{D1})\right)\right), \end{aligned} \tag{9}$$

where $\mathbf{F}_D$ has C_2 channels and a spatial resolution of $H/2\times W/2$. Both convolutions use padding 1.

The downsampled features are subsequently processed by the 9×9 Gaussian module. Its output is added to $\mathbf{F}_D$ and normalized according to Eq. (10):

$$\mathbf{F}_G = \mathrm{BN}_R\left(\mathbf{F}_D + \mathrm{Gaussian}_{9,0.5}(\mathbf{F}_D)\right). \tag{10}$$

This stage includes two residual additions: the first combines the input with $\mathbf{R}_G(\mathbf{F}_D)$ inside the Gaussian module before Conv_Extra, and the second adds the module output to $\mathbf{F}_D$.

DRFD_LoG first expands the features from C_2 to $2C_2$ channels through a depthwise convolution. A 5×5 Gaussian-initialized module then processes the expanded features, followed by residual addition and batch normalization, as described in Eq. (11):

$$\begin{aligned} \mathbf{Z} &= \mathrm{DWConv}^{C_2\to2C_2}_{3\times3,s=1,g=C_2}(\mathbf{F}_G), \\ \mathbf{Z}_G &= \mathrm{Gaussian}_{5,0.5}(\mathbf{Z}), \\ \mathbf{Z}_e &= \mathrm{BN}_g\left(\mathbf{Z} + \mathbf{Z}_G\right). \end{aligned} \tag{11}$$

The resulting feature map $\mathbf{Z}_e$ is passed to two parallel downsampling branches. The convolutional branch applies a stride-2 depthwise convolution, followed by SiLU activation and batch normalization. The pooling branch applies stride-2 max pooling followed by batch normalization. Their outputs are concatenated along the channel dimension and projected by a 1×1 convolution, as expressed in Eq. (12):

$$\begin{aligned} \mathbf{F}_c &= \mathrm{BN}_c\left(\mathrm{SiLU}\left(\mathrm{DWConv}_{3\times3,s=2,g=2C_2}(\mathbf{Z}_e)\right)\right), \\ \mathbf{F}_m &= \mathrm{BN}_m\left(\mathrm{MaxPool}_{3\times3,s=2,p=1}(\mathbf{Z}_e)\right), \\ \mathbf{F}_{\mathrm{out}} &= \mathrm{Conv}^{4C_2\to2C_2}_{1\times1}\left(\mathrm{Concat}_c[\mathbf{F}_c,\mathbf{F}_m]\right). \end{aligned} \tag{12}$$

Each branch produces $2C_2$ channels at a spatial resolution of $H/4 \times W/4$. Concatenation yields $4C_2$ channels, which the projection reduces to $2C_2$.

With $2C_2 = \mathrm{stem_dim}$, the two downsampling stages produce the stride-4 output defined in Eq. (13):

$$\mathbf{F}_{\mathrm{LoGStem}} = \mathrm{DRFD_LoG}(\mathbf{F}_G) \in \mathbb{R}^{B\times\mathrm{stem_dim}\times H/4\times W/4}. \tag{13}$$

These RGB P2 features feed both the additional P2 detection branch and the P3-centered multi-scale aggregation pathway. By combining LoG- and Gaussian-initialized learnable filters with residual processing and dual-branch downsampling, LoGStem provides shallow structural representations for subsequent cross-modal fusion and detection.

3.3.3 P3 Cross-Modal Feature Fusion and P2 Detection Branch

The baseline YOLO11n-MM explicitly fuses RGB and infrared features at P4 and P5 and performs detection at three scales (P3–P5). PVMF-YOLO extends direct cross-modal fusion to P3 and adds a P2 detection branch to make greater use of high-resolution features for small-target detection. Figure 7 illustrates these modifications.

At P3, both backbones produce feature maps with a spatial stride of 8. The corresponding RGB and infrared features are concatenated along the channel dimension and processed by a C3k2 block, as formulated in Eq. (14):

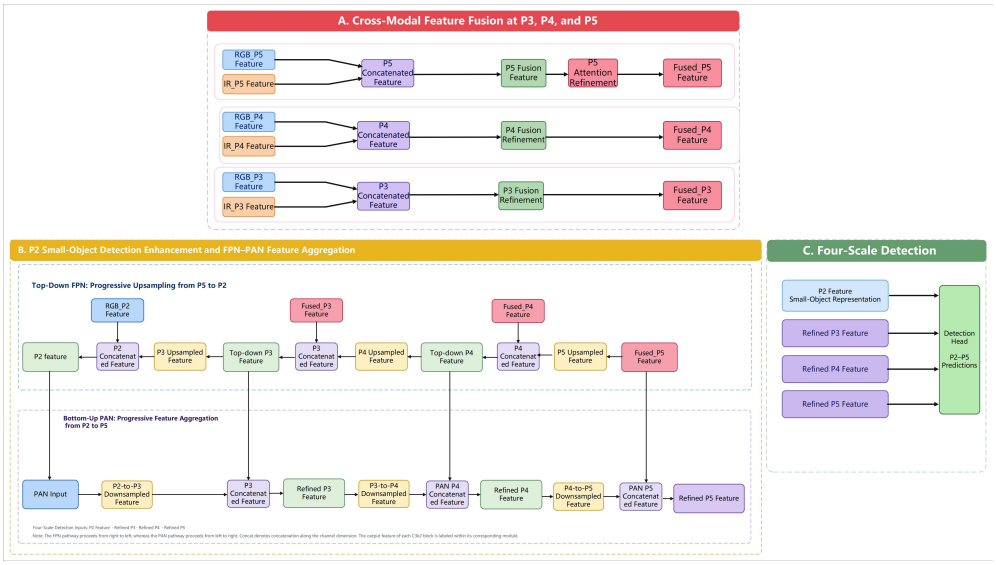


Fig. 7 Architecture of P3 cross-modal feature fusion and the P2 detection branch

$$\mathbf{F}_3^f = \mathrm{C3k2}\left(\mathrm{Concat}_c\left[\mathbf{F}_3^{\mathrm{RGB}}, \mathbf{F}_3^{\mathrm{IR}}\right]\right), \tag{14}$$

where $\mathbf{F}_3^{\mathrm{RGB}}$ and $\mathbf{F}_3^{\mathrm{IR}}$ are the modality-specific P3 features, Concat_c denotes channel-wise concatenation, and $\mathbf{F}_3^f$ is the fused representation. This operation introduces direct interaction between visual and thermal features at a higher spatial resolution than the baseline fusion levels. Fusion at P4 and P5 retains the baseline design, with C3k2 processing at both levels and additional C2PSA refinement at P5.

In the complete PVMF-YOLO architecture, the SOEP–RFPN neck aggregates the fused features to produce an enhanced P3 representation, $\mathbf{E}_3$ (Section 3.3.4). To construct the P2 detection feature, $\mathbf{E}_3$ is upsampled using SNI, concatenated with the RGB P2 features extracted by LoGStem, and processed by a C3k2 block, as expressed in Eq. (15):

$$\mathbf{D}_2 = \mathrm{C3k2}\left(\mathrm{Concat}_c\left[\mathrm{SNI}_2\left(\mathbf{E}_3\right), \mathbf{F}_2^{\mathrm{RGB}}\right]\right), \tag{15}$$

where SNI_2 denotes twofold spatial upsampling, $\mathbf{F}_2^{\mathrm{RGB}}$ is the RGB P2 backbone feature, and $\mathbf{D}_2$ is the resulting stride-4 detection representation. The P2 branch combines shallow RGB details with the enhanced cross-modal representation from P3. It therefore receives infrared information through $\mathbf{E}_3$ rather than through direct fusion of same-scale RGB and infrared P2 features.

The P2 representation is supplied to the four-scale detection head and serves as the starting point for bottom-up aggregation. GSConvE-based downsampling and subsequent feature fusion propagate information from $\mathbf{D}_2$ to the P3–P5 detection representations, as described in Section 3.3.4. This pathway forms part of the complete model, in which the P2 branch operates together with LoGStem and SOEP–RFPN.

To assess P3 fusion and P2 detection separately, three ablation configurations were constructed with the original convolutional stems and FPN–PAN neck, excluding

LoGStem and SOEP–RFPN. All three configurations retain the baseline P4 and P5 cross-modal fusion operations.

In the P2 Detection Only configuration, direct P3 cross-modal fusion is disabled. The top-down pathway first combines the upsampled fused P5 representation with the fused P4 features. The resulting P4 representation is then upsampled and concatenated with the RGB P3 features to construct the top-down P3 representation. A further upsampling step combines this representation with the RGB P2 features to generate the additional P2 detection feature. The bottom-up PAN pathway subsequently aggregates features across P2–P5, and the resulting representations are supplied to a four-scale detection head.

In the P3 Fusion Only configuration, direct RGB–infrared fusion is added at P3, while the original three-scale detection head (P3–P5) is retained. No P2 detection branch is introduced.

The combined P2–P3 Fusion configuration enables both components. The fused P3 features enter the top-down pathway used to construct the P2 detection representation, with the RGB P2 features retained as the lateral input. Bottom-up PAN aggregation then produces the four-scale features supplied to the detection head. These configurations allow P3 fusion and P2 detection to be evaluated individually and jointly without LoGStem or SOEP–RFPN. Modification B therefore comprises two separately testable components, whose combined implementation corresponds to the P2–P3 Fusion configuration.

3.3.4 SOEP–RFPN-Based Multi-Scale Feature Enhancement and Aggregation

SOEP–RFPN replaces the conventional FPN–PAN neck with a structure that enhances features at P3 and aggregates information across P2–P5. SOEP denotes the Small-Object Enhancement Pyramid developed in this study, whose feature propagation strategy draws on Rethinking FPN (Li 2024). The neck integrates Soft Nearest Neighbor Interpolation (SNI), Space-to-Depth Convolution (SPDConv), CSPOmniKernel, and an enhanced GSConv module (GSConvE) for top-down propagation, high-resolution feature integration, and bottom-up refinement. These components bring together shallow RGB features, cross-modal information, and deeper semantic features at the P2–P5 levels. Figure 8 shows the structure of each component.

In the top-down path, SNI uses nearest-neighbor upsampling with fixed amplitude scaling, as described in (Li 2024). For an input feature map $\mathbf{X}$ and an upsampling factor r, SNI is defined in Eq. (16):

$$\mathrm{SNI}_r(\mathbf{X}) = \frac{1}{r^2}\,\mathrm{Upsample}_{\mathrm{nearest},r}(\mathbf{X}), \tag{16}$$

PVMF-YOLO uses $r = 2$, which gives an amplitude coefficient of $1/4$. SNI is used to upsample features from P5 to P4 and from P4 to P3. It is also used to generate the P2 detection feature from the enhanced P3 representation. This operation keeps nearest-neighbor interpolation and adjusts the amplitude of the resulting features. It has no learnable parameters. The upsampled features are then concatenated with features at the same resolution. The fixed amplitude coefficient is retained in each of these cross-level connections.

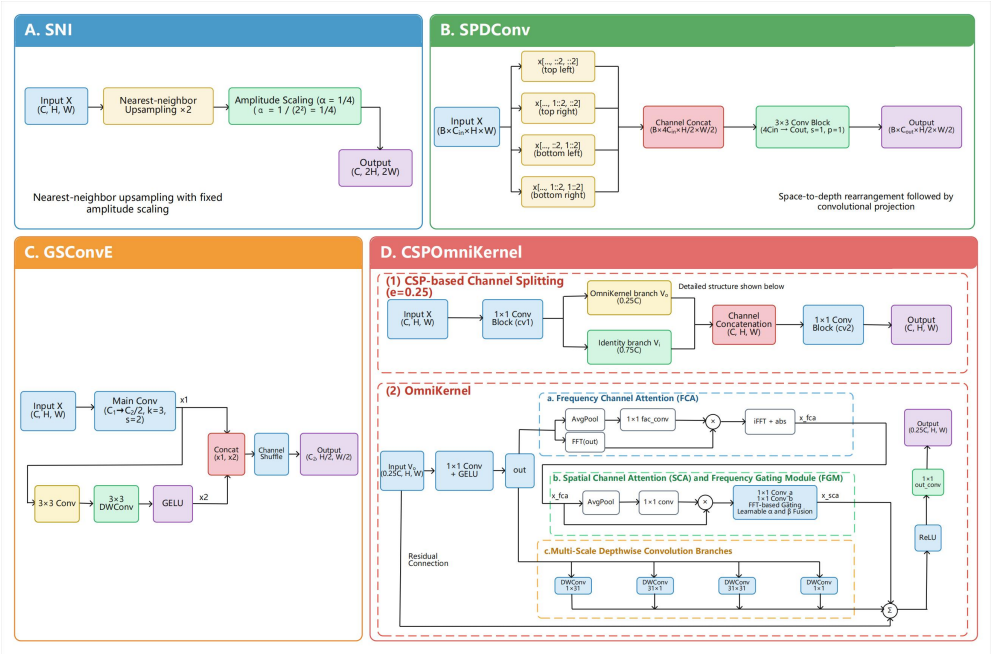


Fig. 8 Key components of SOEP–RFPN: (A) SNI, (B) SPDConv, (C) GSConvE, and (D) CSPOmniKernel

SPDConv (Sunkara and Luo 2023) transfers RGB P2 information to the P3 aggregation stage by rearranging spatial samples into channels before convolutional projection, as formulated in Eq. (17):

$$\mathrm{SPDConv}(\mathbf{X}) = \mathrm{Conv}_{3\times3}\left(\mathrm{SpaceToDepth}_2(\mathbf{X})\right), \tag{17}$$

For even spatial dimensions, $\mathrm{SpaceToDepth}_2$ maps an input of size $B \times C \times H \times W$ to $B \times 4C \times H/2 \times W/2$. The input is rearranged into four interleaved spatial sub-maps, which are then stacked along the channel dimension. This operation preserves all input samples. A 3×3 convolution is then applied, followed by batch normalization and SiLU activation. The convolution mixes the channels and adjusts their number. In the proposed neck, SPDConv converts RGB P2 features at stride 4 into stride-8 features for P3 aggregation. Spatial downsampling takes place during the rearrangement, before the convolution.

The P3 enhancement path combines the rearranged RGB P2 features with cross-modal features at P3 and top-down features. Let $\mathbf{F}_2^{\mathrm{RGB}}$ represent the RGB P2 feature map, and let $\mathbf{F}_3^f$, $\mathbf{F}_4^f$, and $\mathbf{F}_5^f$ represent the fused features at P3–P5. The aggregation process is given in Eq. (18):

$$
\begin{aligned}
\mathbf{T}_4 &= \mathrm{C3k2}\left(\mathrm{Concat}_c\left[\mathrm{SNI}_2(\mathbf{F}_5^f), \mathbf{F}_4^f\right]\right), \\
\mathbf{Q}_3 &= \mathrm{Concat}_c\left[\mathrm{SPDConv}(\mathbf{F}_2^{\mathrm{RGB}}), \mathrm{SNI}_2(\mathbf{T}_4), \mathbf{F}_3^f\right], \\
\mathbf{E}_3 &= \mathrm{C3k2}\left(\mathrm{CSPOmniKernel}(\mathbf{Q}_3)\right),
\end{aligned}
\tag{18}
$$

Here, $\mathbf{T}_4$ represents the top-down P4 feature, $\mathbf{Q}_3$ denotes the concatenated P3 input, and $\mathbf{E}_3$ represents the enhanced P3 output. The three inputs to $\mathbf{Q}_3$ provide complementary information. The RGB P2 pathway provides fine spatial details, the top-down P4 pathway carries deeper contextual features, and the fused P3 pathway provides visual and thermal features at the same scale. CSPOmniKernel and C3k2 process these inputs at P3 before the detection features are constructed.

CSPOmniKernel combines the Cross Stage Partial (CSP) principle (Wang et al. 2020) with OmniKernel (Cui et al. 2024). After a 1×1 projection, the features are split into an OmniKernel branch containing 25% of the channels and a bypass branch containing the remaining 75%. Their outputs are concatenated and projected according to Eq. (19):

$$
\begin{gathered}
\mathbf{V} = \mathrm{Conv}_{1\times1}^{(1)}(\mathbf{Q}_3), \quad (\mathbf{V}_o, \mathbf{V}_i) = \mathrm{Split}(\mathbf{V}; 0.25, 0.75), \\
\mathrm{CSPOmniKernel}(\mathbf{Q}_3) = \\
\mathrm{Conv}_{1\times1}^{(2)}\big(\mathrm{Concat}_c[\mathrm{OmniKernel}(\mathbf{V}_o), \mathbf{V}_i]\big).
\end{gathered}
\tag{19}
$$

For a projected channel dimension C, $\mathbf{V}_o$ and $\mathbf{V}_i$ contain $0.25C$ and $0.75C$ channels, respectively. OmniKernel preserves the dimensions of $\mathbf{V}_o$, allowing its output to be concatenated directly with the bypass features. The final projection mixes the two branches. This partial-processing design confines the spatial and frequency-domain operations of OmniKernel to one quarter of the projected channels while preserving a direct propagation path for the remainder.

Within OmniKernel, Frequency Channel Attention (FCA) applies channel-wise modulation in the frequency domain. A 1×1 convolution and GELU activation are first used to produce an intermediate representation $\mathbf{O}$. Global average pooling is then applied, followed by another 1×1 convolution to generate the modulation coefficients. The FCA operations are given in Eq. (20):

$$
\begin{aligned}
\mathbf{O} &= \mathrm{GELU}\left(\mathrm{Conv}_{1\times1}^{\mathrm{in}}(\mathbf{V}_o)\right), \\
\mathbf{A}_f &= \mathrm{Conv}_{1\times1}^{\mathrm{FCA}}\left(\mathrm{AvgPool}(\mathbf{O})\right), \\
\mathbf{X}_{\mathrm{fca}} &= \left|\mathcal{F}^{-1}\left(\mathbf{A}_f \odot \mathcal{F}(\mathbf{O})\right)\right|,
\end{aligned}
\tag{20}
$$

Here, $\mathcal{F}$ denotes the two-dimensional Fourier transform, and $\mathcal{F}^{-1}$ denotes the inverse transform. The symbol $\odot$ represents element-wise multiplication with broadcasting. The pooled coefficients $\mathbf{A}_f$ are then broadcast across the frequency components of each channel. Taking the magnitude after inverse transformation yields the real-valued feature map $\mathbf{X}_{\mathrm{fca}}$.

Spatial Channel Attention (SCA) is then used to refine the FCA output. The resulting features are passed to a Frequency Gating Module (FGM), where two 1×1 projections are combined as shown in Eq. (21):

$$
\begin{aligned}
\mathbf{A}_s &= \mathrm{Conv}_{1\times1}^{\mathrm{SCA}}\left(\mathrm{AvgPool}(\mathbf{X}_{\mathrm{fca}})\right), \\
\mathbf{U} &= \mathbf{A}_s \odot \mathbf{X}_{\mathrm{fca}}, \\
\mathbf{U}_1 &= \mathrm{Conv}_{1\times1}^{\mathrm{FGM},1}(\mathbf{U}), \\
\mathbf{U}_2 &= \mathrm{Conv}_{1\times1}^{\mathrm{FGM},2}(\mathbf{U}), \\
\mathbf{X}_{\mathrm{sca}} &= \boldsymbol{\alpha} \odot \left|\mathcal{F}^{-1}\left(\mathbf{U}_1 \odot \mathcal{F}(\mathbf{U}_2)\right)\right| \\
&\quad + \boldsymbol{\beta} \odot \mathbf{U},
\end{aligned}
\tag{21}
$$

Both projections are computed from the SCA-modulated feature $\mathbf{U}$, but only $\mathbf{U}_2$ undergoes Fourier transformation. The learnable channel-wise coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are initialized to 0 and 1, respectively. FGM therefore initially returns $\mathbf{U}$ and learns the contribution of the frequency-related term during training. Frequency-domain operations are performed in FP32 for numerical stability during mixed-precision execution.

In parallel, four depthwise convolutions process $\mathbf{O}$ using kernels of 1×31, 31×1, 31×31, and 1×1. The asymmetric kernels capture directional structures, the large square kernel incorporates broader spatial context, and the 1×1 kernel provides a local per-channel response. These outputs are added to the input branch and the SCA–FGM output, as expressed in Eq. (22):

$$
\begin{aligned}
\mathbf{M} &= \mathbf{V}_o + \sum_{k\in\mathcal{K}} \mathrm{DWConv}_k(\mathbf{O}) + \mathbf{X}_{\mathrm{sca}}, \\
\mathrm{OmniKernel}(\mathbf{V}_o) &= \mathrm{Conv}_{1\times1}^{\mathrm{out}}\left(\mathrm{ReLU}(\mathbf{M})\right),
\end{aligned}
\tag{22}
$$

where $\mathcal{K} = \{1\times31, 31\times1, 31\times31, 1\times1\}$. The residual addition retains $\mathbf{V}_o$ while incorporating the spatial and frequency-processing responses. ReLU and a final 1×1 projection produce the OmniKernel output, with all operations confined to the selected CSP channel group.

The enhanced P3 representation $\mathbf{E}_3$ is upsampled and combined with the RGB P2 features to construct $\mathbf{D}_2$, as described in Section 3.3.3. This feature map brings together high-resolution RGB details and cross-modal context and forms the input to the bottom-up pathway. Progressive downsampling is performed by GSConvE, which extends GSConv (Li et al. 2024) with an additional standard convolution in the auxiliary branch. Its operations are defined in Eq. (23):

$$
\begin{aligned}
\mathbf{X}_1 &= \mathrm{ConvBNAct}_{3\times3,s=2}(\mathbf{X}), \\
\mathbf{X}_2 &= \mathrm{GELU}\big(\mathrm{DWConv}_{3\times3,s=1} \\
&\quad \big(\mathrm{Conv}_{3\times3,s=1}(\mathbf{X}_1)\big)\big), \\
&\quad \mathrm{GSConvE}(\mathbf{X}) = \\
&\mathrm{ChannelShuffle}_2(\mathrm{Concat}_c[\mathbf{X}_1, \mathbf{X}_2]).
\end{aligned}
\tag{23}
$$

Here, ConvBNAct comprises convolution, batch normalization, and SiLU activation. The main branch uses a 3×3 stride-2 convolution to generate half of the required output channels. The auxiliary branch processes $\mathbf{X}_1$ through a standard 3×3 convolution and a depthwise 3×3 convolution, with no normalization or activation between

them, followed by GELU. The two outputs have identical spatial dimensions and channel counts. They are concatenated and passed through ChannelShuffle_2 to mix the two channel groups.

At each bottom-up stage, GSConvE downsamples the preceding detection feature, which is then concatenated with the corresponding P3, P4, or P5 representation and processed by C3k2, as formulated in Eq. (24):

$$\begin{aligned}
\mathbf{D}_3 &= \text{C3k2}\left(\text{Concat}_c\left[\text{GSConvE}_{s=2}(\mathbf{D}_2), \mathbf{E}_3\right]\right), \\
\mathbf{D}_4 &= \text{C3k2}\left(\text{Concat}_c\left[\text{GSConvE}_{s=2}(\mathbf{D}_3), \mathbf{T}_4\right]\right), \\
\mathbf{D}_5 &= \text{C3k2}\left(\text{Concat}_c\left[\text{GSConvE}_{s=2}(\mathbf{D}_4), \mathbf{F}_5^f\right]\right),
\end{aligned} \tag{24}$$

The lateral inputs are the enhanced feature $\mathbf{E}_3$ at P3, the top-down representation $\mathbf{T}_4$ at P4, and the fused cross-modal feature $\mathbf{F}_5^f$ at P5. The P5 stage reuses the existing fused representation without introducing another cross-modal fusion operation in the neck. The resulting feature maps $\mathbf{D}_2$, $\mathbf{D}_3$, $\mathbf{D}_4$, and $\mathbf{D}_5$ have strides of 4, 8, 16, and 32, respectively, and feed the four-scale detection head. Through P3-centered enhancement and bidirectional aggregation, SOEP–RFPN connects shallow RGB details with cross-modal features and deeper semantic context.

4 Experiments and Results

4.1 Experimental Setup and Performance Evaluation

4.1.1 Experimental Setup and Training Configuration

All experiments were conducted on the DM-PV dataset using a customized object detection framework based on Ultralytics. Table 2 shows the hardware and software configurations used in the experiments. The same configurations were used throughout all experiments.

Table 2 Hardware and software configuration of the experimental platform

Component	Configuration
Operating system	Windows
GPU	NVIDIA GeForce RTX 5060 Ti
GPU memory	16 GB
System memory	32 GB
Python	3.11.9
PyTorch	2.12.0
CUDA (PyTorch build)	13.0
Detection framework	Customized Ultralytics

All experiments used the training, validation, and test sets described in Section 3.1. Each RGB–infrared pair remained in the same subset. The YOLO11n-MM baseline, all ablation variants, and PVMF-YOLO were trained with the same main hyperparameters and training schedule, as listed in Table 3. These settings were kept unchanged to ensure a fair comparison of the different model modifications.

Table 3 Main training hyperparameters for the YOLO11-based experiments

Hyperparameter	Configuration
Model scale	YOLO11n
Input resolution	640×640
Training epochs	150
Batch size	4
Optimizer	AdamW
Initial learning rate	0.001
First-moment coefficient	0.9
Weight decay	0.0005
Final learning-rate factor	0.01
Warm-up epochs	3
Learning-rate schedule	Linear decay
Random seed	0
AMP	Disabled
Data-loading workers	4

During training, we used Mosaic augmentation with a probability of 1.0, horizontal flipping with a probability of 0.5, and random translation with a factor of 0.1. HSV augmentation was also applied with hue, saturation, and value factors of 0.015, 0.7, and 0.4, respectively. MixUp and Copy-Paste were disabled. The same geometric transformations were applied to both RGB and infrared images in each pair to maintain their spatial correspondence. HSV augmentation was applied only to RGB images and was not used for infrared images. All augmentations were performed online during training, without changing the stored image pairs.

We selected checkpoints based on validation fitness, calculated as $0.1\,\mathrm{mAP}_{50} + 0.9\,\mathrm{mAP}_{50:95}$. For each model, the checkpoint with the highest validation fitness was evaluated on the held-out test set. Test-set results were not used to optimize the models or select checkpoints.

4.1.2 Evaluation Metrics

We evaluated detection performance using Precision, Recall, F1-score, mAP_{50}, and $\mathrm{mAP}_{50:95}$. We also used the number of parameters (Params) and giga floating-point

operations (GFLOPs) to assess model complexity. These metrics provide a combined assessment of detection performance, model size, and computational cost.

Precision and Recall are defined in Eq. (25):

$$\begin{aligned} P &= \frac{TP}{TP + FP}, \\ R &= \frac{TP}{TP + FN}, \end{aligned} \tag{25}$$

where TP, FP, and FN denote true positives, false positives, and false negatives, respectively. Precision and Recall are reported as the average values across the six inspection classes. The F1-score is then calculated as the harmonic mean of these two values, as shown in Eq. (26):

$$F_1 = \frac{2PR}{P + R}. \tag{26}$$

This calculation differs from averaging the F1-scores computed separately for each class.

For class c, average precision (AP) is calculated from the precision–recall curve at a given intersection-over-union (IoU) threshold τ, as shown in Eq. (27):

$$AP_c(\tau) = \int_0^1 P_c(R; \tau)\, dR, \tag{27}$$

Here, $P_c(R; \tau)$ represents precision at recall R for class c and IoU threshold τ. For the six inspection classes ($C = 6$), mAP_{50} is calculated by averaging the AP values across all classes at an IoU threshold of 0.50. $\mathrm{mAP}_{50:95}$ is calculated by averaging AP across all classes and ten IoU thresholds, ranging from 0.50 to 0.95 in steps of 0.05. Both metrics are defined in Eq. (28):

$$\begin{aligned} \mathrm{mAP}_{50} &= \frac{1}{C} \sum_{c=1}^{C} AP_c(0.50), \\ \mathrm{mAP}_{50:95} &= \frac{1}{10C} \sum_{k=0}^{9} \sum_{c=1}^{C} AP_c(0.50 + 0.05k), \end{aligned} \tag{28}$$

By including higher IoU thresholds, $\mathrm{mAP}_{50:95}$ provides a stricter assessment of bounding-box localization than mAP_{50}. Params denotes the total number of model parameters, and GFLOPs estimates theoretical computation at an input resolution of 640×640. These complexity indicators do not directly measure inference latency or throughput.

4.1.3 Training and Validation Convergence Analysis

Optimization behavior was examined for three configurations: the YOLO11n-MM baseline, the model combining LoGStem with P3 fusion and P2 detection (A+B), and the complete PVMF-YOLO model. Figure 9 presents their training and validation box-regression and classification losses over 150 epochs.

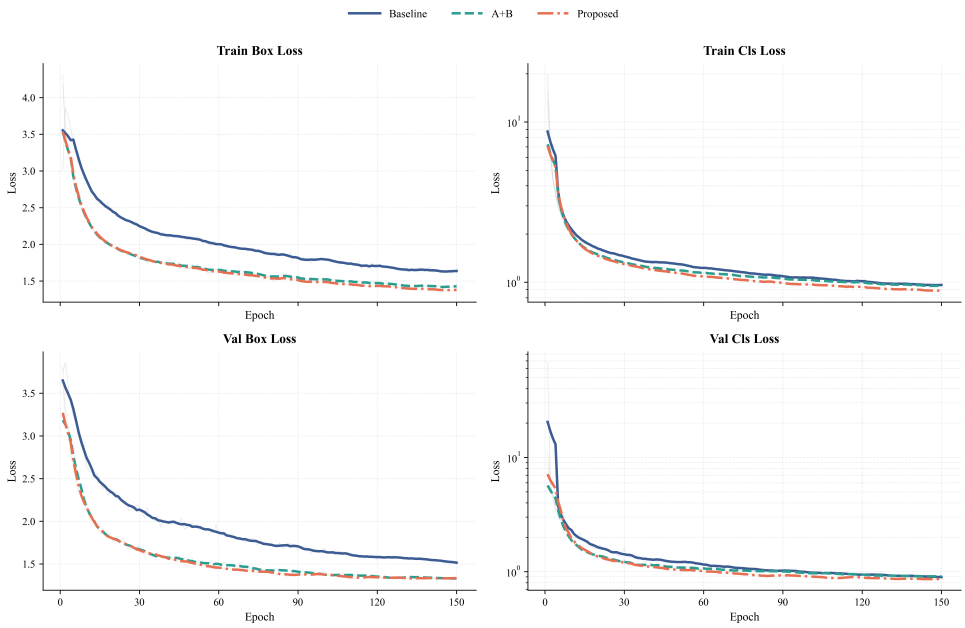


Fig. 9 Training and validation box-regression and classification losses for the YOLO11n-MM baseline, A+B, and PVMF-YOLO over 150 epochs

Losses decrease overall for all three models. A+B and PVMF-YOLO show a faster initial reduction in box-regression loss than the baseline and maintain lower validation box-regression losses during later epochs. These trends are consistent with improved fitting of the regression objective under the shared training settings, although lower validation loss alone does not establish better test-set localization. Classification losses also decline, with smaller differences between the models toward the end of training.

Figure 10 shows the corresponding validation Precision, Recall, mAP_{50}, and $mAP_{50:95}$.

The validation metrics improve quickly during the early stages of training and then remain relatively stable. A+B and PVMF-YOLO generally achieve higher values than the baseline across the reported metrics, while following similar convergence trends to one another. These curves show how validation performance changes during training and are used to select checkpoints. The following sections report detection results on the held-out test set and evaluate the contribution of each modification.

4.2 Comparison with Existing Detection Models

We conducted comparative experiments on the held-out DM-PV test set to compare PVMF-YOLO with other YOLO-based detectors. We also evaluated three input configurations: RGB-only, infrared-only, and dual-modal inputs. All results are reported using the metrics defined in Section 4.1.2.

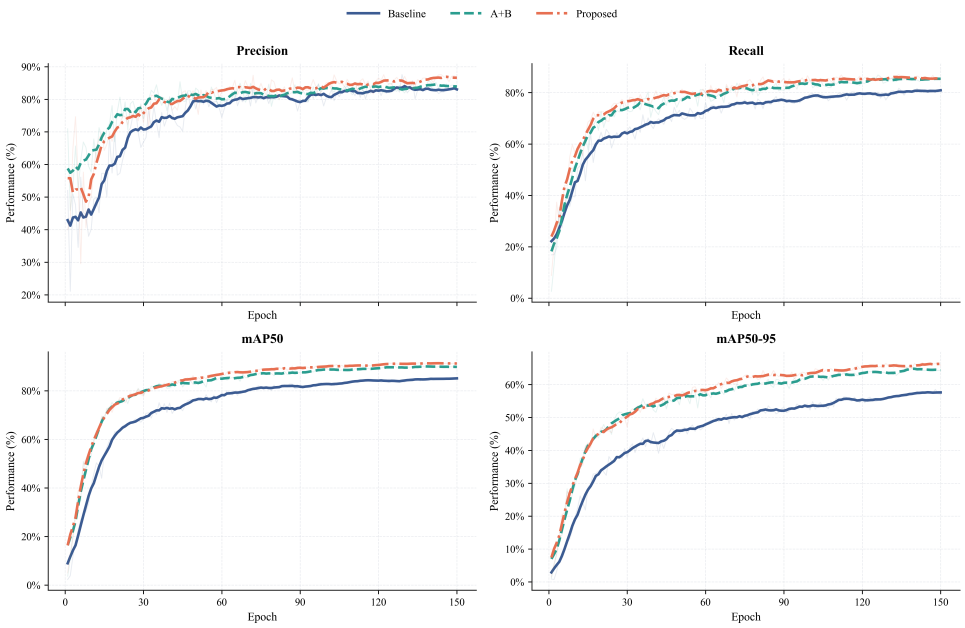


Fig. 10 Validation Precision, Recall, mAP_{50}, and $mAP_{50:95}$ for the YOLO11n-MM baseline, A+B, and PVMF-YOLO over 150 training epochs

4.2.1 Comparison with Dual-Modal Detection Models

PVMF-YOLO was compared with seven dual-modal YOLO variants, including the YOLO11n-MM baseline. The references identify the underlying YOLO architectures; the -MM entries denote the dual-modal adaptations evaluated in this study. Table 4 reports their detection performance and model complexity.

PVMF-YOLO achieves the highest values across all five detection metrics among the evaluated models. Compared with the YOLO11n-MM baseline, it improves mAP_{50} from 0.851 to 0.906 and $mAP_{50:95}$ from 0.591 to 0.645, corresponding to gains of 5.5 and 5.4 percentage points, respectively. YOLOv8n-MM achieves a similar Precision of 0.864, compared with 0.868 for PVMF-YOLO, but has lower Recall, F1-score, and both mAP values. These results show improved detection performance for the complete PVMF-YOLO architecture under the adopted evaluation protocol.

The improvements are accompanied by increased computational cost. Relative to the baseline, the parameter count rises from 4.97 M to 5.91 M, while computation increases from 11.22 to 35.51 GFLOPs. PVMF-YOLO has the highest GFLOPs among the models in Table 4, reflecting its emphasis on detection performance. Section 4.3 evaluates the contribution of each modification. Section 4.5 examines their effects on computational cost.

Table 4 Performance comparison of dual-modal detection models on the DM-PV test set

Model	P	R	F1	mAP_{50}	$mAP_{50:95}$	Params (M)	GFLOPs
PVMF-YOLO	**0.868**	**0.862**	**0.865**	**0.906**	**0.645**	5.91	35.51
YOLO11n-MM (Baseline) (Jocher and Qiu 2024)	0.803	0.809	0.806	0.851	0.591	4.97	11.22
YOLOv5n-MM (Jocher 2022)	0.815	0.797	0.806	0.836	0.567	4.39	11.28
YOLOv6n-MM (Li et al. 2022)	0.776	0.779	0.777	0.806	0.528	8.34	22.72
YOLOv8n-MM (Jocher et al. 2023)	0.864	0.806	0.834	0.853	0.582	5.28	13.18
YOLOv10n-MM (Wang et al. 2024a)	0.758	0.758	0.758	0.817	0.566	4.71	11.95
YOLO12n-MM (Tian et al. 2025)	0.792	0.762	0.777	0.811	0.550	4.73	12.70
YOLO26n-MM (Jocher et al. 2026)	0.718	0.747	0.732	0.779	0.539	4.79	10.46

Note: P denotes Precision; R denotes Recall. Bold values indicate the highest value for each detection metric among the evaluated models. Params and GFLOPs characterize model complexity.

4.2.2 Comparison Among RGB-Only, IR-Only, and Dual-Modal Detection

RGB-only, infrared-only, and dual-modal configurations were evaluated for seven YOLO architectures: YOLOv5n (Jocher 2022), YOLOv6n (Li et al. 2022), YOLOv8n (Jocher et al. 2023), YOLOv10n (Wang et al. 2024a), YOLO11n (Jocher and Qiu 2024), YOLO12n (Tian et al. 2025), and YOLO26n (Jocher et al. 2026). Table 5 presents the results on the DM-PV test set.

All seven architectures achieve higher F1-scores, mAP_{50}, and $mAP_{50:95}$ with dual-modal inputs than with RGB or infrared inputs alone. Compared with RGB-only detection, the gains range from 3.8 to 10.4 percentage points for mAP_{50} and from 0.9 to 9.6 percentage points for $mAP_{50:95}$. Compared with IR-only detection, the improvements range from 4.1 to 9.1 and from 2.0 to 7.9 percentage points, respectively. These metrics show consistent improvements across all evaluated architectures, although the gains vary in magnitude.

YOLO11n shows a similar pattern. Its dual-modal configuration achieves mAP_{50} and $mAP_{50:95}$ values of 0.851 and 0.591, respectively. The corresponding values for RGB-only detection are 0.747 and 0.495, while those for IR-only detection are 0.760 and 0.512. Neither RGB-only nor IR-only detection achieves the best results across all architectures and evaluation metrics. This difference is consistent with the information

Table 5 Performance comparison of YOLO-based detectors using different input modalities on the DM-PV test set

Model	Modality	P	R	F1	mAP_{50}	$mAP_{50:95}$	Params (M)	GFLOPs
YOLO11n (Jocher and Qiu 2024)	RGB–IR	0.803	0.809	0.806	0.851	0.591	4.97	11.22
	RGB	0.756	0.732	0.744	0.747	0.495	2.58	6.47
	IR	0.743	0.761	0.752	0.760	0.512	2.58	6.47
YOLOv5n (Jocher 2022)	RGB–IR	0.815	0.797	0.806	0.836	0.567	4.39	11.28
	RGB	0.746	0.774	0.760	0.776	0.498	2.51	7.42
	IR	0.766	0.737	0.751	0.771	0.516	2.51	7.42
YOLOv6n (Li et al. 2022)	RGB–IR	0.776	0.779	0.777	0.806	0.528	8.34	22.72
	RGB	0.752	0.755	0.753	0.766	0.519	4.23	11.79
	IR	0.745	0.722	0.733	0.742	0.508	4.23	11.79
YOLOv8n (Jocher et al. 2023)	RGB–IR	0.864	0.806	0.834	0.853	0.582	5.28	13.18
	RGB	0.776	0.778	0.777	0.798	0.538	3.01	8.18
	IR	0.785	0.747	0.766	0.774	0.522	3.01	8.18
YOLOv10n (Wang et al. 2024a)	RGB–IR	0.758	0.758	0.758	0.817	0.566	4.71	11.95
	RGB	0.706	0.727	0.716	0.722	0.477	2.27	6.68
	IR	0.667	0.724	0.694	0.736	0.499	2.27	6.68
YOLO12n (Tian et al. 2025)	RGB–IR	0.792	0.762	0.777	0.811	0.550	4.73	12.70
	RGB	0.754	0.728	0.741	0.747	0.497	2.56	7.39
	IR	0.724	0.745	0.734	0.753	0.501	2.56	7.39
YOLO26n (Jocher et al. 2026)	RGB–IR	0.718	0.747	0.732	0.779	0.539	4.79	10.46
	RGB	0.727	0.713	0.720	0.741	0.475	2.41	5.71
	IR	0.714	0.726	0.720	0.738	0.491	2.41	5.71

Note: RGB–IR denotes dual-modal detection, RGB denotes visible-light-only detection, and IR denotes infrared-only detection.

provided by the two modalities. RGB images show surface appearance and structural details, while infrared images provide thermal information.

These comparisons consider differences in both the input modality and the network structure. The observed gains therefore represent the combined effects of multi-modal integration and architectural adaptation; they do not isolate the contribution of the infrared input. Improvements also do not extend to every individual metric. For YOLO26n, dual-modal Precision is 0.718, slightly below the RGB-only value of 0.727, despite higher Recall, F1-score, and both mAP values. In addition, every dual-modal configuration requires more parameters and computation than its single-modal counterparts. Section 4.5 examines this performance–complexity trade-off in more detail.

The consistent gains in F1-score and both mAP metrics support RGB–infrared detection within the evaluated setting. The ablation experiments in Section 4.3 further examine which architectural modifications contribute to the performance of PVMF-YOLO.

4.3 Ablation Studies

Ablation experiments were conducted using YOLO11n-MM as the baseline to assess the individual and combined effects of three modifications: (A) LoGStem-based shallow feature enhancement, (B) P3 cross-modal fusion with an additional P2 detection branch, and (C) SOEP–RFPN-based multi-scale feature enhancement and aggregation. The experiments first evaluated all eight combinations of A, B, and C, then examined the two components of modification B separately. All configurations were trained under the same settings and evaluated on the held-out DM-PV test set.

4.3.1 Ablation of the Architectural Modifications

Table 6 summarizes the results for the eight configurations, and Fig. 11 compares their detection metrics.

Table 6 Ablation results of the proposed architectural modifications on the DM-PV test set

Configuration	P	R	F1	mAP_{50}	$mAP_{50:95}$	Params (M)	GFLOPs
Baseline	0.803	0.809	0.806	0.851	0.591	4.97	11.22
Baseline+A	0.825	0.832	0.828	0.862	0.609	5.01	20.90
Baseline+B	0.810	0.835	0.822	0.874	0.621	5.17	16.73
Baseline+C	0.814	0.790	0.802	0.845	0.571	5.40	17.10
Baseline+A+B	0.858	0.842	0.850	0.896	**0.648**	5.21	26.41
Baseline+A+C	0.833	0.824	0.828	0.871	0.614	5.44	26.78
Baseline+B+C	0.853	0.846	0.849	0.894	0.633	5.87	25.83
PVMF-YOLO (A+B+C)	**0.868**	**0.862**	**0.865**	**0.906**	0.645	5.91	35.51

Note: A denotes LoGStem; B denotes P3 cross-modal fusion and the additional P2 detection branch; C denotes SOEP–RFPN. Bold values indicate the highest value for each detection metric among the evaluated configurations.

The three modifications show different effects when tested individually. LoGStem (A) improves all five detection metrics compared with the baseline. This result shows that it is useful for shallow feature extraction in this setting. Modification B achieves the largest individual improvements in both mAP metrics. It increases mAP_{50} from 0.851 to 0.874 and $mAP_{50:95}$ from 0.591 to 0.621. SOEP–RFPN (C), however, increases Precision but reduces Recall, F1-score, mAP_{50}, and $mAP_{50:95}$ when used alone. This shows that adding this module alone does not consistently improve the baseline model.

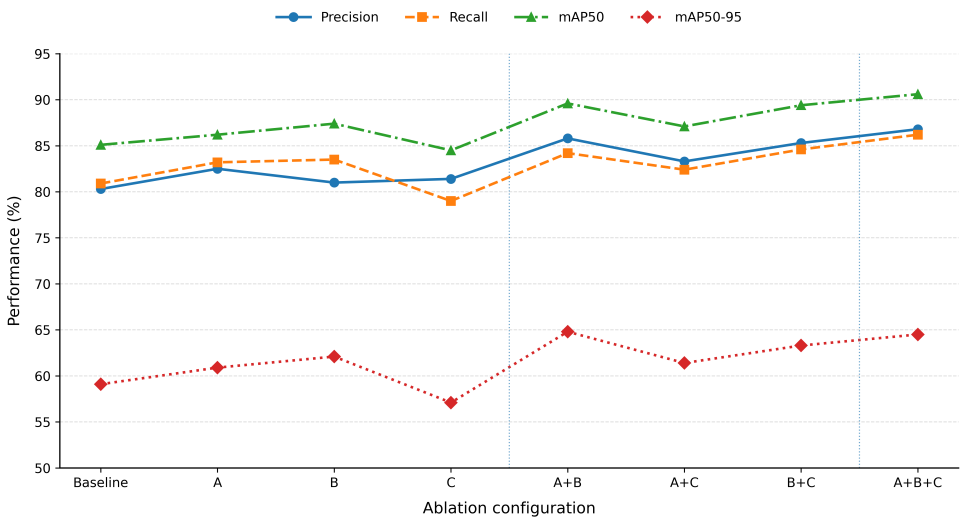


Fig. 11 Performance comparison of the eight ablation configurations on the DM-PV test set

The results show that the effects of the modifications depend on how they are combined. Among the eight configurations, A+B achieves the highest $mAP_{50:95}$ of 0.648. Although C reduces both mAP metrics when used alone, adding it to A increases mAP_{50} from 0.862 to 0.871 and $mAP_{50:95}$ from 0.609 to 0.614. A similar improvement is observed when C is added to B, with the two values increasing from 0.874 and 0.621 to 0.894 and 0.633, respectively. These results show that the effect of SOEP–RFPN depends on the other feature extraction and fusion components in the model.

The full PVMF-YOLO model achieves the highest Precision, Recall, F1-score, and mAP_{50}. However, its $mAP_{50:95}$ of 0.645 is slightly lower than the 0.648 achieved by A+B. Compared with A+B, adding C increases Precision, Recall, F1-score, and mAP_{50} by 1.0, 2.0, 1.5, and 1.0 percentage points, respectively. At the same time, $mAP_{50:95}$ decreases by 0.3 percentage points. Therefore, the full model performs best on four of the five metrics, while A+B achieves a slightly higher $mAP_{50:95}$.

These performance differences also come with changes in computational cost. PVMF-YOLO has 5.91 M parameters and requires 35.51 GFLOPs, compared with 4.97 M parameters and 11.22 GFLOPs for the baseline. The A+B configuration has 5.21 M parameters and requires 26.41 GFLOPs. It also achieves a slightly higher $mAP_{50:95}$ than the full model. Therefore, A+B is a more efficient choice when computational cost and $mAP_{50:95}$ are the main concerns. Section 4.5 provides further analysis of these differences.

4.3.2 Independent Ablation of P3 Fusion and P2 Detection

Modification B includes two components: direct P3 cross-modal fusion and an additional P2 detection branch. To evaluate their effects, we added each component to

the baseline separately and then combined them. All configurations retained the original convolutional stems, FPN–PAN neck, and P4/P5 cross-modal fusion. Neither LoGStem nor SOEP–RFPN was used. Table 7 shows the results.

Table 7 Independent ablation results of P3 cross-modal fusion and P2 detection on the DM-PV test set

Configuration	P3 Fusion	P2 Detection	P	R	F1	mAP_{50}	$mAP_{50:95}$	Params (M)	GFLOPs
Baseline	–	–	0.803	0.809	0.806	0.851	0.591	4.97	11.22
P2 Detection Only	–	✓	0.808	0.817	0.812	0.864	0.597	5.04	15.10
P3 Fusion Only	✓	–	0.836	0.799	0.817	0.848	0.594	5.09	12.84
P2 + P3	✓	✓	0.810	0.835	0.822	0.874	0.621	5.17	16.73

Note: P3 Fusion denotes additional direct RGB–IR feature fusion at P3; P2 Detection denotes the additional high-resolution P2 detection branch. All configurations retain the baseline P4/P5 cross-modal fusion. P2 + P3 corresponds to Baseline+B in Table 6. A dash indicates that the corresponding component is disabled.

Adding only the P2 detection branch improves all five detection metrics compared with the baseline. Specifically, mAP_{50} increases from 0.851 to 0.864, and $mAP_{50:95}$ increases from 0.591 to 0.597. P3 fusion alone has a different effect. Precision increases from 0.803 to 0.836, while Recall decreases from 0.809 to 0.799. F1-score and $mAP_{50:95}$ improve slightly, but mAP_{50} decreases from 0.851 to 0.848. These results show that P3 fusion alone does not improve all detection metrics.

Combining the two components gives the best Recall, F1-score, mAP_{50}, and $mAP_{50:95}$ among the configurations listed in Table 7. The combined model achieves an F1-score of 0.822, mAP_{50} of 0.874, and $mAP_{50:95}$ of 0.621. Compared with P2 Detection Only, adding P3 fusion increases $mAP_{50:95}$ by 2.4 percentage points. Precision remains below that of P3 Fusion Only, but the combined configuration achieves higher Recall and stronger results on both mAP metrics. This pattern supports a complementary effect between high-resolution prediction and direct cross-modal fusion in the evaluated setting.

The combined configuration requires 5.17 M parameters and 16.73 GFLOPs, compared with 5.04 M and 15.10 GFLOPs for P2 Detection Only and 5.09 M and 12.84 GFLOPs for P3 Fusion Only. Jointly enabling the components therefore improves F1-score and both mAP metrics at additional computational cost. Although this design aims to detect small defects, the overall results do not show how the improvements vary across different object sizes.

4.4 Qualitative Detection Results and Error Analysis

We compared the detection results of the YOLO11n-MM baseline, A+B, and PVMF-YOLO on selected samples from the DM-PV test set. The examples include bird

droppings, single hotspot, and weed occlusion, covering different defect types and background conditions. Figure 12 shows the paired RGB–infrared images, ground-truth annotations, and predictions from the three models.

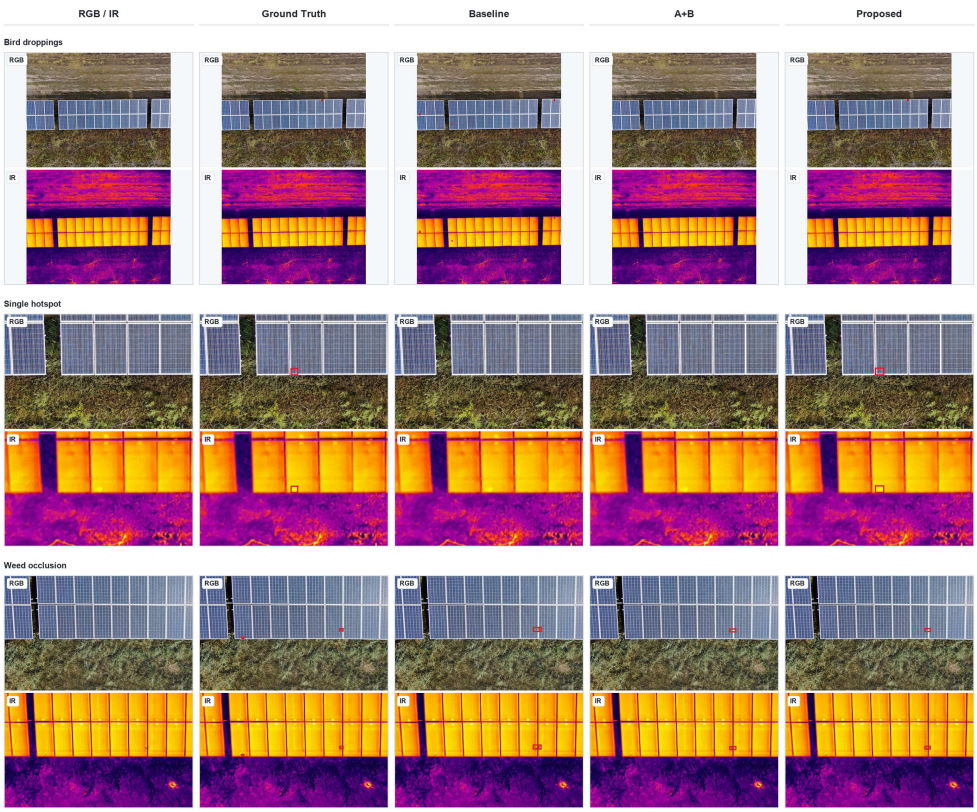


Fig. 12 Qualitative comparison of the YOLO11n-MM baseline, A+B, and PVMF-YOLO on representative DM-PV test samples covering bird droppings, single hotspot, and weed occlusion. Paired RGB–infrared inputs and ground-truth annotations are shown alongside the predictions

In the single-hotspot example, both the baseline and A+B miss a small defect near the lower edge of a PV module, while PVMF-YOLO detects it at the correct location. The other examples show how the three models perform on different defect types and under different background conditions. These results are consistent with the quantitative results in Table 6. PVMF-YOLO achieves higher Recall and mAP_{50} than the baseline. However, its $mAP_{50:95}$ is slightly lower than that of A+B (0.645 versus 0.648).

Figure 13 shows three typical cases of missed detections, false positives, and bounding-box localization errors. Each case shows a full-scene image and enlarged RGB–infrared regions, making it easier to compare the model predictions with the ground-truth annotations.

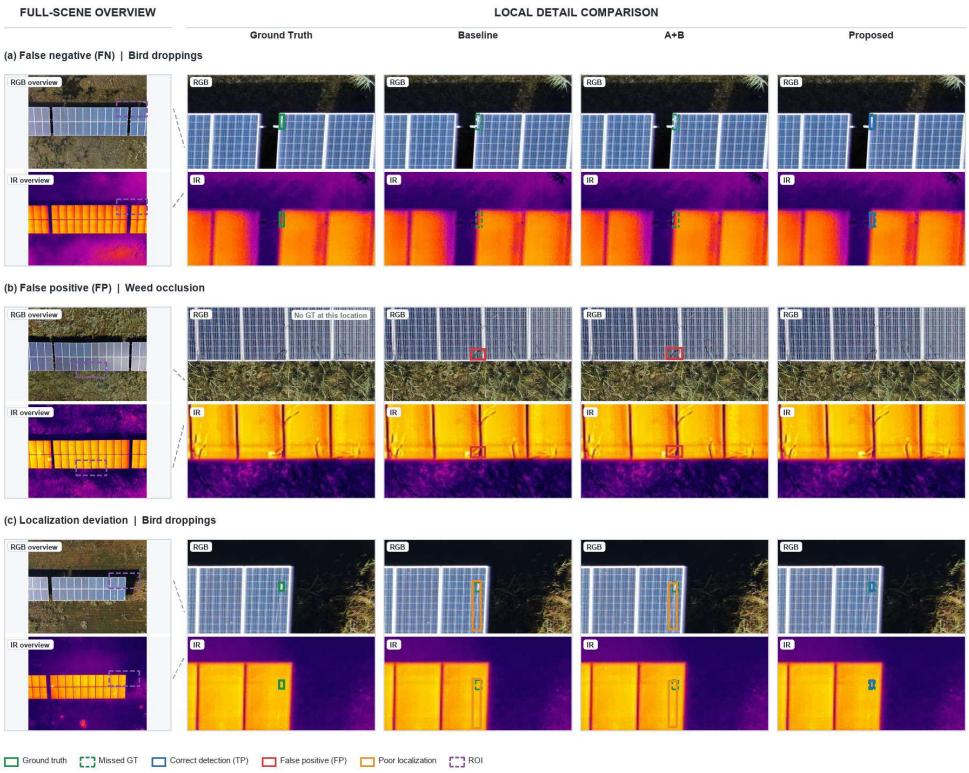


Fig. 13 Representative cases involving (a) missed bird-dropping defects, (b) false positives in a weed-occlusion scene, and (c) bounding-box localization errors for bird droppings. Each case includes a full-scene view and enlarged RGB–infrared regions showing the ground-truth annotations and model predictions

In case (a), the baseline and A+B fail to detect the annotated bird-dropping defect, while PVMF-YOLO detects it successfully. In case (b), the baseline and A+B produce false positives in the weed-occlusion region, but these false detections are not observed in the PVMF-YOLO results. In case (c), the predicted bounding boxes from the baseline and A+B extend beyond the annotated defect. In comparison, PVMF-YOLO produces a bounding box that more closely matches the ground-truth annotation.

In these examples, PVMF-YOLO detects some targets missed by the other models, avoids some false positives, and produces bounding boxes that better match the ground-truth annotations. These observations illustrate specific improvements but do not establish their frequency across the test set. The slightly higher $mAP_{50:95}$ of A+B also shows that the full model does not consistently improve localization performance. This result is consistent with the ablation results in Section 4.3.

4.5 Model Complexity Analysis

Computational cost is also important in PV inspection, along with detection accuracy. We evaluated model complexity using the number of parameters (Params) and

theoretical floating-point operations (GFLOPs) at an input resolution of 640×640. Table 6 shows these results for the eight ablation configurations. Figure 14 provides a visual comparison of their computational complexity.

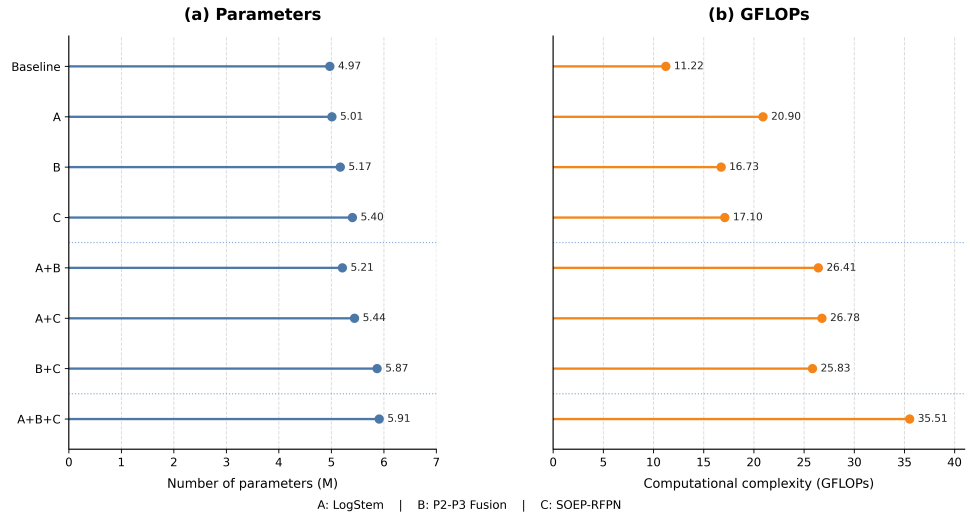


Fig. 14 Parameter counts and theoretical computational costs of the eight ablation configurations

PVMF-YOLO uses 18.9% more parameters and requires 216.5% more GFLOPs than the baseline. Its theoretical computational cost is therefore approximately 3.17 times that of YOLO11n-MM, despite a smaller proportional increase in parameter count. The individual modifications also differ in their computational requirements. A increases Params by 0.8% and GFLOPs by 86.3%; B increases them by 4.0% and 49.1%, respectively; and C increases them by 8.7% and 52.4%. In particular, the small parameter increase associated with A does not translate into a similarly small computational overhead, illustrating why parameter count alone is insufficient to characterize computational demand. C also introduces additional computation while reducing both mAP metrics when used alone, as shown in Section 4.3.

Adding C to A+B increases the number of parameters by 13.4% and GFLOPs by 34.5%. The full model achieves the highest Precision, Recall, F1-score, and mAP_{50} among the ablation configurations. However, A+B achieves a slightly higher $mAP_{50:95}$. Therefore, A+B is a lower-cost option when computational resources and performance at stricter IoU thresholds are important. The complete model provides gains on the other four detection metrics at additional computational cost. Its GFLOPs are also the highest among the dual-modal detectors in Table 4, reflecting the computational demands of its emphasis on detection performance.

Params and GFLOPs describe model size and theoretical computation, but they do not directly measure inference latency, throughput, or runtime memory usage. These deployment characteristics depend on the target hardware and implementation.

As no inference-time benchmarks were conducted in this study, real-time deployment capability cannot be established from the reported complexity measures alone.

5 Discussion

PVMF-YOLO improves PV defect detection over the YOLO11n-MM baseline by combining shallow feature enhancement, direct cross-modal fusion, and multi-scale aggregation. The ablation results show, however, that these modifications contribute differently and that their effects depend on the configuration in which they are used. The complete model achieves the highest Precision, Recall, F1-score, and mAP_{50} among the ablation configurations, while A+B obtains a slightly higher $mAP_{50:95}$.

Across the seven YOLO architectures, dual-modal configurations achieve higher F1-scores, mAP_{50}, and $mAP_{50:95}$ than the corresponding RGB-only and IR-only configurations. This result may be related to the different types of information provided by the two modalities. RGB images provide surface appearance and structural details, while infrared images provide thermal information. However, this comparison cannot isolate the effect of infrared inputs because the dual-modal models also have different network structures. Therefore, the observed improvements may result from both multimodal inputs and the corresponding changes in network design.

The ablation experiments show the effects of the proposed modifications. LoGStem improves all five detection metrics when used alone. This result supports its use for shallow feature extraction in this setting. Modification B combines P3 fusion with P2 detection and achieves the largest individual gains in both mAP metrics. The separate experiments in Section 4.3.2 further show the effects of these two components. P2 detection alone improves all five metrics compared with the baseline. In contrast, P3 fusion alone improves Precision but reduces Recall and slightly decreases mAP_{50}. When combined, the two components achieve higher Recall, F1-score, mAP_{50}, and $mAP_{50:95}$ than either component alone. However, further evaluation by object size is needed to determine how much of these improvements comes from detecting small defects.

The effect of SOEP–RFPN depends on the other components in the model. When used alone, it improves Precision but reduces Recall, F1-score, mAP_{50}, and $mAP_{50:95}$. However, when combined with either A or B, it improves both mAP metrics. Adding it to A+B raises mAP_{50} from 0.896 to 0.906 but lowers $mAP_{50:95}$ from 0.648 to 0.645. This result shows that better performance at an IoU threshold of 0.50 does not always lead to better performance at the higher IoU thresholds used to calculate $mAP_{50:95}$. It also shows that adding more feature aggregation does not improve every model configuration or evaluation metric. Since repeated training runs were not conducted, the stability of these differences remains uncertain, especially the small difference in $mAP_{50:95}$.

The computational results provide further information about the performance gains. A+B requires 26.41 GFLOPs, while the full model requires 35.51 GFLOPs. A+B also achieves a higher $mAP_{50:95}$. Therefore, A+B is a lower-cost option when computational cost and localization performance at higher IoU thresholds are important. Compared with A+B, the full PVMF-YOLO model achieves better results in

the other four detection metrics but requires more computation. However, Params and GFLOPs alone are not enough to determine whether the model is suitable for deployment. These measures do not include inference latency, throughput, or runtime memory requirements.

Several limitations should be considered when interpreting the results. First, the evaluation was conducted only on the DM-PV dataset. The model has not been tested on other PV installations, imaging devices, or environmental conditions. Second, the dataset was divided at the paired-sample level because the available metadata did not support scene-level splitting. Although file identifiers were checked to avoid overlap between subsets, some scenes may still appear in more than one subset. This may affect the test results. In addition, the released RGB–infrared pairs have spatial misalignment, and no further geometric registration was performed in this study. These differences may reduce the correspondence between the two modalities, especially for small defects. Finally, class-wise and object-size-specific performance analyses, repeated-run statistical evaluation, and hardware-based inference measurements were not conducted.

Future work will assess the framework on additional PV datasets and use scene-separated evaluation where suitable acquisition metadata are available. Alignment-aware fusion will be investigated to improve tolerance to residual spatial discrepancies, together with more efficient shallow feature extraction and multi-scale aggregation. Class-wise and object-size-specific analyses will help identify where the modifications are most effective. Repeated training runs and hardware-based benchmarks will also be used to assess performance variability and practical deployment requirements.

6 Conclusion

This study proposed PVMF-YOLO, a YOLO11n-based RGB–infrared detector that integrates LoGStem, P3 cross-modal fusion with an additional P2 detection branch, and SOEP–RFPN-based multi-scale aggregation for photovoltaic defect detection.

On the DM-PV dataset, PVMF-YOLO achieves a Precision of 0.868, a Recall of 0.862, an F1-score of 0.865, an mAP_{50} of 0.906, and an $mAP_{50:95}$ of 0.645. Compared with the baseline, all five metrics improve by 6.5, 5.3, 5.9, 5.5, and 5.4 percentage points, respectively. Across the seven YOLO architectures, dual-modal configurations achieve higher F1-scores, mAP_{50}, and $mAP_{50:95}$ than the corresponding RGB-only and IR-only configurations. The ablation results show that Modification B provides the largest individual gains in both mAP metrics. The effect of SOEP–RFPN depends on how it is combined with the other components. The full model achieves the highest Precision, Recall, F1-score, and mAP_{50}. However, A+B achieves a slightly higher $mAP_{50:95}$ than PVMF-YOLO (0.648 versus 0.645).

These improvements require higher computational resources. PVMF-YOLO has 5.91 M parameters and requires 35.51 GFLOPs, compared with 4.97 M parameters and 11.22 GFLOPs for the baseline. Future work will focus on evaluating cross-dataset generalization, developing fusion methods that account for misalignment, studying the effects of P3 fusion and P2 detection on defects of different sizes, conducting repeated training runs, and testing inference efficiency on different hardware.

References

Akram MW, Li G, Jin Y, et al (2020) Automatic detection of photovoltaic module defects in infrared images with isolated and develop-model transfer deep learning. Solar Energy 198:175–186. https://doi.org/10.1016/j.solener.2020.01.055

Akram MW, Li G, Jin Y, et al (2022) Failures of photovoltaic modules and their detection: A review. Applied Energy 313:118822. https://doi.org/10.1016/j.apenergy.2022.118822

Bommes L, Hoffmann M, Buerhop-Lutz C, et al (2022) Anomaly detection in IR images of PV modules using supervised contrastive learning. Progress in Photovoltaics: Research and Applications 30:597–614. https://doi.org/10.1002/pip.3518

Cao Y, Pang D, Yan Y, et al (2023) A photovoltaic surface defect detection method for building based on deep learning. Journal of Building Engineering 70:106375. https://doi.org/10.1016/j.jobe.2023.106375

Cui Y, Ren W, Knoll A (2024) Omni-kernel network for image restoration. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp 1426–1434, https://doi.org/10.1609/aaai.v38i2.27907

Fan Y, Duan K, Zhang Y (2026) An efficient dual-branch multi-scale approach for photovoltaic defect detection in electroluminescence images. Journal of Real-Time Image Processing 23:13. https://doi.org/10.1007/s11554-025-01808-6

Fang Q, Wang Z (2022) Cross-modality attentive feature fusion for object detection in multispectral remote sensing imagery. Pattern Recognition 130:108786. https://doi.org/10.1016/j.patcog.2022.108786

Gao W, Liao G, Ma S, et al (2022) Unified information fusion network for multi-modal RGB-D and RGB-T salient object detection. IEEE Transactions on Circuits and Systems for Video Technology 32(4):2091–2106. https://doi.org/10.1109/TCSVT.2021.3082939

Gong L, Wang Y, Lv Y, et al (2025) A lightweight and efficient model for photovoltaic panel defect detection in complex environments. Engineering Applications of Artificial Intelligence 159:111764. https://doi.org/10.1016/j.engappai.2025.111764

Høiaas I, Grujic K, Imenes AG, et al (2022) Inspection and condition monitoring of large-scale photovoltaic power plants: A review of imaging technologies. Renewable and Sustainable Energy Reviews 161:112353. https://doi.org/10.1016/j.rser.2022.112353

Hong F, Song J, Meng H, et al (2022) A novel framework on intelligent detection for module defects of PV plant combining the visible and infrared images. Solar Energy

236:406–416. https://doi.org/10.1016/j.solener.2022.03.018

Huang W, Zhao J, Wei R, et al (2026) MRFP-TDG: A detection transformer with hybrid encoder and position-aware decoder for photovoltaic cell defect detection. Information Sciences 735:123092. https://doi.org/10.1016/j.ins.2026.123092

Jocher G (2022) YOLOv5 by ultralytics. Software, version 7.0, https://doi.org/10.5281/zenodo.3908559, URL https://github.com/ultralytics/yolov5

Jocher G, Qiu J (2024) Ultralytics YOLO11. Ultralytics software release, version 11.0.0, URL https://github.com/ultralytics/ultralytics, official documentation: https://docs.ultralytics.com/models/yolo11/. Ultralytics states that no formal research paper has been published for YOLO11.

Jocher G, Chaurasia A, Qiu J (2023) Ultralytics YOLOv8. Software, version 8.0.0, URL https://github.com/ultralytics/ultralytics

Jocher G, Qiu J, Liu M, et al (2026) Ultralytics YOLO26: Unified real-time end-to-end vision models. arXiv preprint arXiv:2606.03748, https://doi.org/10.48550/arXiv.2606.03748, URL https://arxiv.org/abs/2606.03748

Kang N, Hu W, Xu R, et al (2026) Multi-modal photovoltaic anomaly detection via synergistic graph-contrastive regularization. Energy 361:141986. https://doi.org/10.1016/j.energy.2026.141986

Kellil N, Aissat A, Mellit A (2023) Fault diagnosis of photovoltaic modules using deep neural networks and infrared images under algerian climatic conditions. Energy 263:125902. https://doi.org/10.1016/j.energy.2022.125902

Korovin A, Vasilev A, Egorov F, et al (2023) Anomaly detection in electroluminescence images of heterojunction solar cells. Solar Energy 259:130–136. https://doi.org/10.1016/j.solener.2023.04.059

Kuo CFJ, Chen SH, Huang CY (2023) Automatic detection, classification and localization of defects in large photovoltaic plants using unmanned aerial vehicles (UAV) based infrared (IR) and RGB imaging. Energy Conversion and Management 276:116495. https://doi.org/10.1016/j.enconman.2022.116495

Lai YS, Hsieh CC, Liao TW, et al (2025) Deep learning-based automatic defect detection of photovoltaic modules in infrared, electroluminescence, and red–green–blue images. Energy Conversion and Management 332:119783. https://doi.org/10.1016/j.enconman.2025.119783

Li B, Delpha C, Diallo D, et al (2021) Application of artificial neural networks to photovoltaic fault detection and diagnosis: A review. Renewable and Sustainable Energy Reviews 138:110512. https://doi.org/10.1016/j.rser.2020.110512

Li C, Li L, Jiang H, et al (2022) YOLOv6: A single-stage object detection framework for industrial applications. arXiv preprint arXiv:2209.02976, https://doi.org/10.48550/arXiv.2209.02976, URL https://arxiv.org/abs/2209.02976

Li H (2024) Rethinking features-fused-pyramid-neck for object detection. In: Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part LXVII, Lecture Notes in Computer Science, vol 15125. Springer, Cham, pp 74–90, https://doi.org/10.1007/978-3-031-72855-6_5

Li H, Li J, Wei H, et al (2024) Slim-neck by GSConv: A lightweight design for real-time detector architectures. Journal of Real-Time Image Processing 21(3):62. https://doi.org/10.1007/s11554-024-01436-6

Lin TY, Dollár P, Girshick R, et al (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 2117–2125, https://doi.org/10.1109/CVPR.2017.106

Liu S, Qi L, Qin H, et al (2018) Path aggregation network for instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 8759–8768, https://doi.org/10.1109/CVPR.2018.00913

Ma Y, Chai L, Mao S, et al (2026) Learning modality knowledge with proxy for RGB-infrared object detection. Pattern Recognition 176:113227. https://doi.org/10.1016/j.patcog.2026.113227

Marr D, Hildreth EC (1980) Theory of edge detection. Proceedings of the Royal Society of London Series B, Biological Sciences 207(1167):187–217. https://doi.org/10.1098/rspb.1980.0020

Mellit A (2022) An embedded solution for fault detection and diagnosis of photovoltaic modules using thermographic images and deep convolutional neural networks. Engineering Applications of Artificial Intelligence 116:105459. https://doi.org/10.1016/j.engappai.2022.105459

Mohanty SR, Maruf MU, Singh V, et al (2025) Machine learning approaches for automatic defect detection in photovoltaic systems. Solar Energy 298:113672. https://doi.org/10.1016/j.solener.2025.113672

Ozturk E, Ogliari E, Sakwa M, et al (2024) Photovoltaic modules fault detection, power output, and parameter estimation: A deep learning approach based on electroluminescence images. Energy Conversion and Management 319:118866. https://doi.org/10.1016/j.enconman.2024.118866

Qu Z, Zang J, Li L, et al (2024) Biomimetic model of photovoltaic cell defect detection based on mimic vision. Applied Energy 376:124033. https://doi.org/10.1016/j.apenergy.2024.124033

Rahman MM, Khan I, Alameh K (2021) Potential measurement techniques for photovoltaic module failure diagnosis: A review. Renewable and Sustainable Energy Reviews 151:111532. https://doi.org/10.1016/j.rser.2021.111532

Ramachandran ME, Balachandran GB, Velladurai P, et al (2025) Deep learning for solar PV fault classification using RGB imaging and comparison of preprocessing techniques. Solar Energy 301:113959. https://doi.org/10.1016/j.solener.2025.113959

Sabry AH, Bıyıkoğlu A, Camdali U (2026) A comprehensive review of unmanned aerial vehicle-based thermal imaging and deep learning for PV power plant anomaly detection and performance assessment. Engineering Applications of Artificial Intelligence 163:113070. https://doi.org/10.1016/j.engappai.2025.113070

Shen J, Chen Y, Liu Y, et al (2024) ICAFusion: Iterative cross-attention guided feature fusion for multispectral object detection. Pattern Recognition 145:109913. https://doi.org/10.1016/j.patcog.2023.109913

Shen J, Zhan H, Zuo X, et al (2026) IRDFusion: Iterative relation-map difference guided feature fusion for multispectral object detection. Pattern Recognition 176:113189. https://doi.org/10.1016/j.patcog.2026.113189

Song K, Zhao Y, Huang L, et al (2023) RGB-T image analysis technology and application: A survey. Engineering Applications of Artificial Intelligence 120:105919. https://doi.org/10.1016/j.engappai.2023.105919

Su B, Chen H, Chen P, et al (2021) Deep learning-based solar-cell manufacturing defect detection with complementary attention network. IEEE Transactions on Industrial Informatics 17(6):4084–4095. https://doi.org/10.1109/TII.2020.3008021

Su B, Chen H, Zhou Z (2022) BAF-detector: An efficient CNN-based detector for photovoltaic cell defect detection. IEEE Transactions on Industrial Electronics 69(3):3161–3171. https://doi.org/10.1109/TIE.2021.3070507

Sunkara R, Luo T (2023) No more strided convolutions or pooling: A new CNN building block for low-resolution images and small objects. In: Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2022, Lecture Notes in Computer Science, vol 13715. Springer, Cham, pp 443–459, https://doi.org/10.1007/978-3-031-26409-2_27, conference held in 2022

Tan M, Pang R, Le QV (2020) Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 10781–10790, https://doi.org/10.1109/CVPR42600.2020.01079

Tang W, Yang Q, Hu X, et al (2022) Deep learning-based linear defects detection system for large-scale photovoltaic plants based on an edge-cloud computing infrastructure. Solar Energy 231:527–535. https://doi.org/10.1016/j.solener.2021.11.01

Tian Y, Ye Q, Doermann D (2025) YOLOv12: Attention-centric real-time object detectors. arXiv preprint arXiv:2502.12524, https://doi.org/10.48550/arXiv.2502.12524, URL https://arxiv.org/abs/2502.12524

Wang A, Chen H, Liu L, et al (2024a) YOLOv10: Real-time end-to-end object detection. arXiv preprint arXiv:2405.14458, https://doi.org/10.48550/arXiv.2405.14458, URL https://arxiv.org/abs/2405.14458

Wang B, Chen Q, Wang M, et al (2024b) PVF-10: A high-resolution unmanned aerial vehicle thermal infrared image dataset for fine-grained photovoltaic fault classification. Applied Energy 376:124187. https://doi.org/10.1016/j.apenergy.2024.124187

Wang CY, Liao HYM, Wu YH, et al (2020) CSPNet: A new backbone that can enhance learning capability of CNN. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp 1571–1580, https://doi.org/10.1109/CVPRW50498.2020.00203

Wang H, Chen H, Wang B, et al (2022a) High-efficiency low-power microdefect detection in photovoltaic cells via a field programmable gate array-accelerated dual-flow network. Applied Energy 318:119203. https://doi.org/10.1016/j.apenergy.2022.119203

Wang H, Song K, Huang L, et al (2023) Thermal images-aware guided early fusion network for cross-illumination RGB-T salient object detection. Engineering Applications of Artificial Intelligence 118:105640. https://doi.org/10.1016/j.engappai.2022.105640

Wang Q, Chi Y, Shen T, et al (2022b) Improving RGB-infrared object detection by reducing cross-modality redundancy. Remote Sensing 14(9):2020. https://doi.org/10.3390/rs14092020

Wang Y, Li L, Sun Y, et al (2021) Adaptive automatic solar cell defect detection and classification based on absolute electroluminescence imaging. Energy 229:120606. https://doi.org/10.1016/j.energy.2021.120606

Wang Y, Hu Z, Peng X, et al (2026) Infrared–visible multi-sensor fusion for UAV photovoltaic defect detection under real-world weak misalignment. Remote Sensing 18(15):2607. https://doi.org/10.3390/rs18152607

Yahya Z, Sebari I, Hajji H, et al (2022) Applied imagery pattern recognition for photovoltaic modules' inspection: A review on methods, challenges and future development. Sustainable Energy Technologies and Assessments 52:102071. https://doi.org/10.1016/j.seta.2022.102071

Yuan M, Wei X (2024) C2Former: Calibrated and complementary transformer for RGB-infrared object detection. IEEE Transactions on Geoscience and Remote Sensing 62:5403712. https://doi.org/10.1109/TGRS.2024.3376819

Yuan M, Shi X, Wang N, et al (2024) Improving RGB-infrared object detection with cascade alignment-guided transformer. Information Fusion 105:102246. https://doi.org/10.1016/j.inffus.2024.102246

Zhang J, Chen X, Wei H, et al (2024a) A lightweight network for photovoltaic cell defect detection in electroluminescence images based on neural architecture search and knowledge distillation. Applied Energy 355:122184. https://doi.org/10.1016/j.apenergy.2023.122184

Zhang J, Yang W, Chen Y, et al (2024b) Fast object detection of anomaly photovoltaic (PV) cells using deep neural networks. Applied Energy 372:123759. https://doi.org/10.1016/j.apenergy.2024.123759

Zhu L, Cheng J, Liu G (2026) A novel deep learning model for defect detection in photovoltaic panels using visible light imaging. Engineering Applications of Artificial Intelligence 163:112895. https://doi.org/10.1016/j.engappai.2025.112895